Издательство «Знание»

ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА И ЕЁ ПРИМЕНЕНИЕ

Новое в жизни, науке, технике

Подписная научно-популярная серия

Издается ежемесячно с 1988 г.

Игра — дело серьезное

1991

9

Новое
в жизни,
науке,
технике

ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА
И ЕЁ ПРИМЕНЕНИЕ

Подписная
научно-
популярная
серия

9/1991

Издается
ежемесячно
с 1988 г.

ИГРА — ДЕЛО СЕРЬЕЗНОЕ

Москва
Издательство
"Знание"
1991

ББК 32.97
И 27

Авторы выпуска

Н.ГЕРМАН — кандидат физико-математических наук, работает в Институте высоких температур АН СССР.
ЖАРИКОВ ЛЕОНИД НИКОЛАЕВИЧ — научный сотрудник, занимается разработкой программного обеспечения АСУ.
ЛЮБУТОВ ОЛЕГ ДМИТРИЕВИЧ — инженер-программист.
ХМЕЛЮК ВЛАДИМИР АРКАДЬЕВИЧ — кандидат технических наук, лауреат Государственной премии СССР, специалист в области космической радиосвязи.
В этом выпуске мы публикуем материалы наших читателей Д.Бровкова, К.Погосовой, Р.Сузи, А.Бугайнова, А.Милюкова, А.Кирика, Л.Жданова, Д.Новикова, В.Зорина, В.Воронихина, А.Туманова.

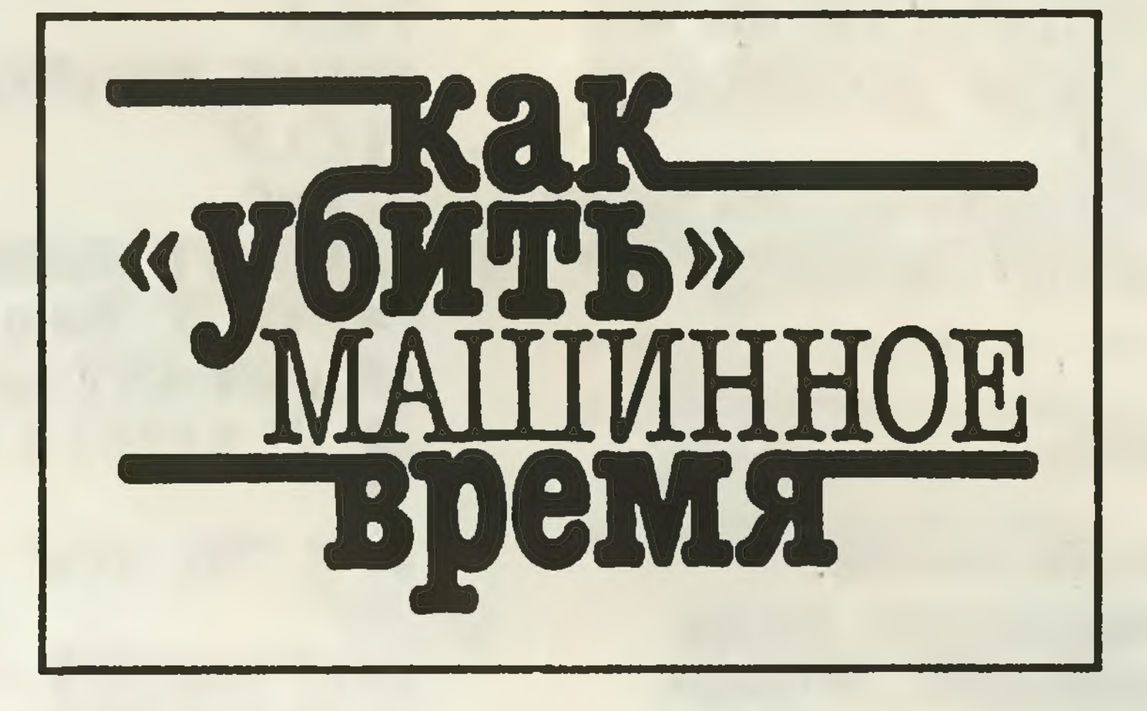
Редактор Б.М.Васильев

ISBN 5-07-002178-8

© Издательство "Знание", 1991 г.

К ЧИТАТЕЛЮ

Сегодня мы удовлетворяем многочисленные запросы наших читателей, имеющих не только много свободного машинного времени, но и просто много свободного времени — сейчас пора летних отпусков. Наша подборка статей поможет вам справиться с проблемой:



С удовольствием сообщаем любителям игр, что не все предложения поместились в один этот выпуск, и еще много компьютерных игр осталось в редакционном портфеле. Все они в дальнейшем найдут место в нашей постоянной рубрике.

Статьи с заключением рецензента проверены, исправлены. Некоторые доработаны. Остальные не проверялись, и потому в последующих выпусках возможна публикация опечаток. Есть такой список и в этом выпуске — внесите правки.

Известно, что много положительных эмоций дает познавательная, хорошо оформленная игра, и не менее интересно снабдить любимую игрушку красивой графикой, музыкальным оформлением, развить ее возможности, наполнить сервисом.

Так как мы считаем, что

то желаем вам не только приятного, но и полезного проведения времени за компьютером.

Д.Бровков
(г.Саратов)

Буквы

Игра написана для машины МС-0511, язык BASIC-УК-НЦ Вильнюс-88. Для БК нужно снизить параметр x, так как игра рассчитана для монитора с 63 символами в строке, т.е. $X_н = X_c/2$. Все остальное менять не требуется, так как BASIC-УК-НЦ является производной от BASIC Вильнюс-86.

Играющий должен нажимать на клавишу, соответствующую "летящему" по экрану символу.

Описание программы

Строка 8 — очистка памяти.
Строка 10 — определение размеров массивов имен (E$), штрафа (R) и класса (A).
Строки 12 — 20 — ввод исходных данных (имя и класс) и запрос на начало.
Строки 23 — 33 — вывод постоянной (на время игры) таблички.
Строки 34 — 50 — собственно игра.
Строка 35 — определение номера строки, по которой будет двигаться символ.
Строка 36 — определение кода символа, которому придется "лететь".
Строки 41 — 42 — формируют задержку с учетом опыта играющего.
Строка 45 — проверка на совпадение нажатой клавиши с символом на экране.
Строка 43 — пробел.
Строка 47 — пустая, прогонка кода символа.
Строка 48 — подсчет штрафных очков: прибавляет количество позиций по X, пройденных символом.
Строки 56 — 88 — вывод таблицы, в которых указано имя, штраф и класс.

```
1 " Программа "БУКВЫ"
2 " ****************
3 " Программа составлена
4 " Студентом 1-го курса
5 " С.П.И., Ф.З.Т. и П.
6 " Бровковым Дмитрием
7 " ******************
8 CLEAR
9 B=1
10 DIM E$ (50), R(50), A(50)
11 CLS
12 C=0
13 INPUT "Ваше имя"; E$ (B)
14 INPUT "Ваш класс (1-5)"; A(B)
15 A(B)=INT (A(B))
16 IF A(B)< 1 OR A(B)> 5 THEN
14
17 ? "Ну что," ; E$ (B);", начнем?
(D/N)"
18 IF INKEY$ ="D" THEN 21
19 IF INKEY$ ="N" THEN 56
20 GOTO 18
21 CLS
22 R(B)=0
23 ? AT(53,3); E$ (B)
24 ? AT(53,5) "Класс-"A(B)
25 ? AT (52,8) "°°°°°°°°°°°° "
26 ? AT (53,10) "Штрафные"
27 ? AT (55,11) "очки"
28 ? AT (54,13) R(B)
29 ? AT (52,15) "°°°°°°°°°°°°"
30 FOR Y=1 TO 20
31 ? AT (51, Y) "°"
32 ? AT (63,Y) "°"
33 NEXT Y
34 FOR I=1 T0 25
35 Y=INT (19*RND(1)+1)
36 S=INT (25*RND (1) +65)
37 IF S=S1 THEN 35
38 A$ =CHR$ (S)
39 FOR X=1 TO 50
40 ? AT(X,Y)A$
41 FOR J=1 TO 50/A(B)
42 NEXT J
43 ? AT (X,Y) "  "
44 S1=S
45 IF INKEY$ = A$ THEN 47
46 NEXT X
47 S=INT (25*RND(1)+65)
48 R(B)=R(B)+X
49 ? AT (54,13)R(B)
50 NEXT I
51 ? AT (10,10) "Вы набрали "R(B)"
штрафных очков."
```

```
52 ? TAB(15) "Сыграем еще? (D/N)";
53 IF INKEY$ = "D" THEN 57
54 IF INKEY$ = "N" THEN 56
55 GOTO 53
56 C=1
57 CLS
58 MIN=2000
59 IF LEN (E$ (B)) <= THEN 61
6O E$ (B)=MID$ (E$ (B), 1,7)
61 FOR J=1 TO B
62 IF MIN R (J) THEN MIN=R(J)
63 NEXT J
64 FOR J=1 TO B
65 IF R(J)=MIN THEN T = J ELSE 76
66 K=R(1)
67 R(1)=R(T)
68 C$ = E$ (1)
69 R(T)=K
70 E$ (1) = E$ (T)
71 E$ (T) = C$
72 L=A(1)
73 A(1)=A(T)
74 A(T)=L
75 GOTO 77
76 NEXT J
77 CLS
78 ? AT (13,1) "==========================================="
79 ? AT (13,2)"!"; TAB(11)"!"; TAB(20)"!"; TAB(33)"!
80 ? AT(13,3)" ! N ! ИМЯ ! ШТРАФ ! КЛАСС !"
81 ? AT(13,4) "!"; TAB (11)"!"; TAB(20)"!"; TAB(33)"!
82 ? AT(13,5)"==========================================="
83 FOR J=1 TO B
84 ? TAB(5)"!"; TAB(11)"!"; TAB (20)"!"; TAB (33)" ! !"
85 ? TAB(5)"!"; J; TAB(11); "! "; E$ (J); TAB(20); "! "; R(J); TAB(33); "! "A(J)" "__!"
86 ? TAB(5)"!"; TAB(11)"!"; TAB(20)"!"; TAB(33)"! !"
87 ? TAB(5)"-----------------------------------"
88 NEXT J
89 B=B+1
90 IF C >0 TNEN 95
91 ? " "
92 ? "Для продолжения игры нажмите на любую клавишу";
93 IF INKEY$ =" " THEN 93
94 GOTO 11
95 END
```

Заключение рецензента

Обучающая игровая программа, по сути, достаточно проста и позволяет изучить латинскую клавиатуру персонального компьютера. Предлагаемая программа без изменений может работать на БК0010.01 в режиме 64 символа в строке.

К сожалению, вариант, предложенный автором, обладает невысоким игровым эффектом вследствие дискретности перемещения символов и слабости обратной связи в случае правильных и неправильных ответов игрока. Есть в тексте программы и неточности. Для правильного вывода таблицы результатов необходимо в строках 78 — 82 заменить число 13 на 5. Окончание строки 85 должно выглядеть так:

85 ... ... A(J) " __ !" ,
где __ — пробел.

Для замедления скорости перемещения символов можно увеличить длительность цикла в строках 41 — 42.

Быстрые горизонтальные дискретные "прыжки" символов утомляют зрение, поэтому предпочтительнее вертикальное или вертикально-горизонтальное перемещение. Лучше двигать не буквы, а слова (по типу игры "Тетрис"), чтобы повысить обучающий эффект, информативность игры и интерес к ней. В этом случае имеется возможность изучения правописания слов и клавиатуры одновременно.

Как правильно заметил автор программы, "здесь можно много чего еще придумать".

Жариков Л.Н.

О.Д.Любутов (г.Ереван)

Игра "Арканоид"

Игра "Арканоид" представляет собой аналог одноименной игры реализованной на компьютере "Ямаха" (MSX-1) и IBM PC. Из-за аппаратных особенностей ЭВМ БК-0010 игра несколько отличается от "фирменных" аналогов.

Игра написана на языке высокого уровня БЕЙСИК с использованием вставок в машинных кодах. Занимает примерно 6,3 кБ программного ОЗУ.

Правила игры

Игрок должен, управляя ракеткой с помощью стрелочных клавиш (<вправо>, <влево>), отражать перемещающийся по полю экрана шарик. В центральной части экрана находится "стена", состоящая из отдельных "кирпичей". Шарик, хаотично перемещаясь по экрану, выбивает те "кирпичи" "из стены", которых он касается (некоторые "кирпичи" могут не выбиваться с первого касания или вообще не выбиваться). При выбивании случайных "кирпичей" шарик изменяет направление своего движения, что затрудняет задачу игрока. Со временем скорость шарика возрастает, что также усложняет игру. Время от времени сверху опускаются разноцветные кубики. При попадании кубика на ракетку игрока могут произойти следующие изменения игровой обстановки.

Если кубик красный, то длина ракетки игрока увеличивается (положительный фактор).

Зеленый — то длина ракетки игрока уменьшается (отрицательный фактор).

Если кубик синий, то скорость шарика уменьшается до начального значения (положительный фактор).

Если кубик полосатый, то игроку добавляется еще одна попытка (положительный фактор).

Первоначально игрок располагает двадцатью попытками. Если игроку не удается отразить шарик и тот выходит за пределы игрового поля, то количество попыток уменьшается на одну, а в случае отсутствия запаса попыток игра заканчивается.

Игрок может изменять направление движения шарика. Если шарик отражается от неподвижной ракетки, то угол его падения равен углу отражения. Если горизонтальные составляющие движений шарика и ракетки совпадают, то угол падения больше угла отражения (то есть полет шарика становится более горизонтальным). Если горизонтальные составляющие движений шарика и ракетки противоположны по направлению, то угол падения меньше угла отражения (полет шарика становится более вертикальным).

Если шарик выбил все "кирпичи" "стены", то очередной этап игры заканчивается и игрок переходит в следующий этап (всего этапов 15). При переходе к следующему этапу количество попыток увеличивается на две.

Игра проста, наглядна, развивает реакцию и глазомер.

Рекомендации

Все комментарии, находящиеся в начале программы, следует исключить. Сначала набрать все строки с операторами DATA (1000 — 1105). Затем набрать следующую программу:

```
1 S!=0
2 FOR I=1 TO 245
3 READ A%
4 S!=S!+A%
5 NEXT I
6 PRINT S!
```

Запустить программу на выполнение. Если в результате появится число 1219230, то все в порядка. Наберите DELETE -6 и продолжайте набор программы. В противном случае следует проверить правильность набора блока оператора DATA.

Описание работы программы

В программе использованы четыре подпрограммы в машинных кодах. Они располагаются в следующих адресах ОЗУ:

USR0 — (37400 — 37446) подпрограмма обеспечивает перемещение

шарика по экрану (входной параметр — адрес ОЗУ экрана, где нужно высветить шарик).

USR! — (37450 — 37476) программа проверки пустого места экрана. Входной параметр — адрес ОЗУ проверяемого места экрана. Выходной параметр — 0, если место занято.

Остальные две подпрограммы (USR2 37500 — 37534 и USR3 37536 — 37562) служат для перемещения "случайной помехи" вверху экрана, перемещения ракетки, попыток и стирания "кирпичей".

В строках 1035 — 1105 закодированы 15 различных конфигураций "стен". При желании пользователь может по своему усмотрению изменить или добавить новые этапы в игру.

Стены закодированы следующим образом.

В каждой строке (1035 — 1105) закодировано по одному этапу. В начале записано число в десятичной форме — это количество выбиваемых "кирпичей" в данном этапе. Следом идет число в восьмеричной форме. В нем закодировано число ударов шарика, необходимое для выбивания "кирпича" данного цвета. Для каждого из трех цветов (начиная с красного) отводится по 4 двоичных разряда (то есть от 0 до 15). Причем если закодировано число 15, то "кирпич" данного цвета считается вообще не выбиваемым. В следующих восьми восьмеричных числах закодирована сама "стена". Кодировка производится по столбцам, сверху, внизу, поцветно. На каждый цвет отводится по 2 бита. Число видимых "кирпичей" не должно быть меньше числа выбиваемых "кирпичей" (первого числа). Пользуясь этими правилами, можно создать любую конфигурацию "стены".

Остальная программа достаточно легко читается — любой желающий без труда сможет в ней разобраться.

```
3 'Игровая программа *АРКАНОИД*
5 'Любутов О.Д. г.Ереван ул. Церетели
5 'д. 46  кв. 38  тел. 42-33-96
5 'Версия 1.2  1989 г.
5 DIM D%(6%,7%),Q%(2%)
10 DRAW "S4;"
15 ? CHR$(&0214)CHR$(&0214)CHR$(&0224)
   CHR$(&0236)
20 GOTO 1215
25 DR%=&044340
30 DD%=1%
35 RR%=5%
40 TO%=1%
45 SL%=0%
50 FP%=0%
55 KP%=13%
60 LT%=0%
65 RESTORE 1005
70 FOR I%=&037400 TO &037562 STEP 2%
75 READ A%
80 POKE I%,A%
85 NEXT I%
90 DEF USR0=&037400
95 DEF USR1=&037450
100 DEF USR2=&037500
105 DEF USR3=&037536
110 FOR I%=&040000 TO &041000 STEP 64%
115 READ A%,A0%,A1%
120 POKE I%,A%
125 POKE I%+2%,A0%
130 POKE I%+6%,A1%
135 NEXT I%
140 FOR I%=&040112 TO &041012 STEP 64%
145 POKE I%,&0177777
150 POKE I%+8%,&052525
155 POKE I%+4%,&0125252
160 POKE I%+12%,&0175245
165 NEXT I%
170 CLS
175 LINE (15%,11%)-(240%,228%),1%,B
180 PAINT (0%,0%),1%
185 COLOR 2%
190 ? AT(14%,23%);"=LOD=";
195 ? AT(4%,0%);"СЧЕТ:";AT(17%,0%);"ПОПЫ
   ТКИ:"
200 LA%=0%
205 KP%=KP%+3%
210 IF LT%=26% THEN 1120
215 GOSUB 840
220 RL%=&075235
225 RP%=&075240
230 IF KP%=1% THEN 1110 ELSE KP%=KP%-1%
235 ? AT(25%,0%);KP%
240 GOSUB 765
245 IZ%=3%
250 KS%=&074636
255 A%=USR0(KS%)
260 GOSUB 715
265 DX%=SGN(RND(1%)-.5!)
270 DY%=-192%
275 SK%=210%
280 KK%=KS%+DX%+DY%
285 IF USR1(KK%)<>0% THEN 335
290 GOTO 320
295 DY%=-DY%
300 KK%=KS%+DX%+DY%
305 IF USR1(KK%)<>0% THEN 335
310 DX%=-DX%
315 DY%=-DY%
320 KK%=KS%+DX%+DY%
325 IF USR1(KK%)<>0% THEN 335
330 DY%=-DY%
335 KK%=KS%+DX%+DY%
340 KS%=KK%
345 A%=USR0(KS%)
350 I$=INKEY$
355 IF I$=CHR$(12%) THEN 535
```

```
360 IF I$<>" " THEN 370
365 IF INKEY$="" THEN 365
370 FOR J%=0% TO SK%
375 NEXT J%
380 TO%=3%-TO%
385 ON TO% GOSUB 680,615
390 GOSUB 680
395 IF PP%=0% THEN 280
400 IF PP%<&074700 THEN 425
405 POKE &037510,&040040
410 A%=USR2(PP%)
415 PP%=0%
420 GOTO 280
425 IF USR1(PP%+&01100)=0% AND PP%>&073000
    THEN 450
430 POKE &037510,CC%
435 PP%=PP%+&0100
440 A%=USR2(PP%)
445 GOTO 280
450 IF CC%<>&040012 THEN 470
455 IZ%=1%
460 GOSUB 715
465 GOTO 405
470 IF CC%<>&040016 THEN 490
475 IZ%=-1%
480 GOSUB 715
485 GOTO 405
490 IF CC%<>&040022 THEN 505
495 SK%=210%
500 GOTO 405
505 KP%=KP%+1%
510 ? AT(25%,0%);KP%
515 GOTO 405
520 BEEP
525 IF SK%>0% THEN SK%=SK%-3%
530 IF KK%<&076100 THEN 545
535 GOSUB 790
540 GOTO 220
545 IF KK%<&043400 OR KK% MOD 64%<4% OR
    KK% MOD 64%>59% THEN 295
550 IF KK%<&074600 THEN 585
555 IF U%=0% THEN 295
560 IF U%=DX% THEN 575
565 IF ABS(DY%+SGN(DY%))<385% THEN DY%=DY%
    +SGN(DY%)*64%
570 GOTO 295
575 IF ABS(DY%-SGN(DY%))>63% THEN DY%=DY%
    -SGN(DY%)*64%
580 GOTO 295
585 IF KK%>&046200 AND KK%<&060600 THEN
    GOSUB 935
590 IF LA%<LT% THEN 605
595 GOSUB 790
600 GOTO 200
605 IF LA% MOD 2%=0% AND PP%=0% THEN
    GOSUB 985
610 GOTO 295
615 OR%=OR%+DD%
620 POKE &037510,&040000+RR%
625 A%=USR2(OR%)
630 IF OR%>&044304 AND OR%<&044371 THEN 655
635 DD%=-DD%
640 RR%=11%-RR%
645 POKE &037510,&040000+RR%
650 A%=USR2(OR%)
655 RETURN
660 LINE (X%,Y%)-@(21%,9%),C%,B
665 PAINT (X%+2%,Y%+2%),C%,C%
670 DRAW "C4NH2R17NE2D5NF2L17N62U5;"
675 RETURN
680 U%=0%
681 IF INP(&0177716,64%)<>0% THEN 710
685 IF INP(&0177662,&0177)=&031 AND RP%<=
    &075270 THEN U%=1%
690 IF INP(&0177662,&0177)=&010 AND RL%>=
    &075205 THEN U%=-1%
695 RP%=RP%+U%
700 RL%=RL%+U%
705 GOSUB 765
710 RETURN
715 SS%=SGN(IZ%)
720 EE%=RP%-RL%+SS%+SS%
725 IF EE%<3% OR EE%>30% THEN 760
730 FOR I%=0% TO IZ% STEP SS%
735 IF RP%>&075270 THEN RL%=RL%-SS% ELSE
    RP%=RP%+SS%
740 GOSUB 765
745 IF RL%<&075205 THEN RP%=RP%+SS% ELSE
    RL%=RL%-SS%
750 GOSUB 765
755 NEXT I%
760 RETURN
765 POKE &037510,&040000
770 A%=USR2(RL%)
775 POKE &037510,&040002
780 A%=USR2(RP%)
785 RETURN
790 A%=USR0(RL%+&0301)
795 A%=(RL%-&075177)*4%
800 A0%=(RP%-RL%+1%)*4%
805 FOR I%=1% TO 100%
810 LINE (A%+RND(1%)*A0%,218%+RND(1%)*9%)-
    @(1%,1%),4%,B
815 NEXT I%
820 FOR I%=RL%-&0300 TO RP%-&0303
825 A%=USR3(I%)
830 NEXT I%
835 RETURN
840 READ LT%,A%
845 Q%(0%)=A%MOD16%
850 A%=A%\16%
855 Q%(1%)=A%MOD16%
860 A%=A%\16%
865 Q%(2%)=A%MOD16%
870 FOR I%=0% TO 7%
875 READ A%
880 FOR J%=0% TO 6%
885 C%=A% MOD 4%
890 IF C%=0% THEN 900
895 D%(J%,I%)=Q%(C%-1%)
900 A%=A%\4%
905 X%=I%*24%+32%
910 Y%=J%*12%+36%
915 GOSUB 660
920 NEXT J%
925 NEXT I%
930 RETURN
935 JJ%=(KK%-&046010)\&01400
```

```
940 IF JJ%=7% THEN JJ%=6%
945 II%=((KK%-8%)MOD64%)\6%
950 IF D%(JJ%,II%)<>15% THEN D%(JJ%,II%)=
    D%(JJ%,II%)-1%
955 IF D%(JJ%,II%)<>0% THEN 980
960 LA%=LA%+1%
965 SL%=SL%+10%
970 ? AT(9%,0%);SL%
975 A%=USR3(&046310+JJ%*&01400+II%*6%)
980 RETURN
985 CC%=INT(RND(1%)*3%+.2!)*4%+&040012
990 PP%=&060710+INT(RND(1%)*44%)
995 RETURN
1000 DATA &075,&0341,&0362,&0353,&0341,&0356,
    &0357,&0351,&0344,&075
1005 DATA &011505,&012700,&040000,&012701,
    &037444,&010537,&037404,&012702,&04,
    &0105010,&0112115,&062705,&0100,&062700,
    &0100,&05302,&01370,&0207,&0125050,&024252
1010 DATA &011502,&012700,&04,&0111201,
    &01005,&062702,&0100,&05300,&01372,&0207,
    &05015,&0207
1015 DATA &011505,&012700,&011,&012701,&00,
    &0112125,&0112125,&0112125,&062705,&075,
    &062701,&075,&05300,&01367,&0207
1020 DATA &011505,&012700,&014,&05025,&05025,
    &05025,&062705,&072,&05300,&01371,&0207
1025 DATA &0100000,&01377,&02520,&0124000,
    &025377,&012524,&0124000,&025377,&054545,
    &0125000,&0125377,&064551,&0125000,
    &0125377,&052525,&0125000,&0125377,&013244
1030 DATA &0124000,&025377,&013224,&0124000,
    &025377,&02520,&0100000,&01377,&02520
1035 DATA 56%,&0421,&034733,&034733,&034733,
    &034733,&034733,&034733,&034733,&034733
1040 DATA 54%,&02023,&012625,&013245,&015351,
    &025472,&025472,&015351,&013245,&012625
1045 DATA 50%,&0521,&037045,&037045,&037045,
    &037045,&025052,&035051,&037245,&037625
1050 DATA 8%,&07421,&037763,&030003,&030103,
    &030623,&030623,&030103,&030003,&037763
1055 DATA 46%,&0421,&012525,&015251,&014011,
    &014311,&014311,&014011,&015251,&012525
1060 DATA 42%,&0442,&012525,&037777,&025252,
    &00,&00,&025252,&037777,&012525
1065 DATA 56%,&022,&023146,&014631,&023146,
    &014631,&023146,&014631,&023146,&014631
1070 DATA 56%,&01423,&012526,&012532,&012553,
    &012557,&013277,&013377,&025777,&027777
1075 DATA 56%,&0446,&035573,&025152,&035573,
    &012525,&012525,&035573,&025152,&035573
1080 DATA 38%,&07421,&033367,&012525,&016733,
    &032527,&035253,&027356,&025252,&035673
1085 DATA 56%,&0521,&012652,&013253,&015257,
    &025277,&025277,&015257,&013253,&012652
1090 DATA 54%,&01422,&05252,&024676,&022676,
    &022652,&022652,&022676,&024676,&05252
1095 DATA 56%,&0437,&023126,&023146,&022546,
    &025252,&037777,&036527,&033777,&036527
1100 DATA 21%,&0437,&012535,&017735,&016735,
    &016735,&016635,&016535,&017775,&012525
1105 DATA 26%,&0777,&035253,&035777,&036537,
    &032007,&036537,&035253,&034013,&037257
1110 ?AT(11%,14%)"КОНЕЦ ИГРЫ"
1115 IF INKEY$="" THEN 1115 ELSE GOTO 25
1120 ?AT(3%,13%)"ВЫ В СОВЕРШЕНСТВЕ ОВЛАДЕЛИ"
1125 ?AT(2%,14%)"МАСТЕРСТВОМ ИГРЫ В АРКАНОИД!
1130 ?AT(4%,15%)"ВАМ ПРИСВАИВАЕТСЯ ЗВАНИЕ"
1135 ?AT(9%,16%)"=АРКАН-МАСТЕР="
1140 ?AT(8%,17%)"АВТОР ЖЕЛАЕТ ВАМ"
1145 ?AT(7%,18%)"ДАЛЬНЕЙШИХ УСПЕХОВ!"
1150 GOTO 1115
1215 RESTORE 1105
1220 GOSUB 340
1225 RESTORE
1230 COLOR 1%
1235 ?AT(2%,1%)"Любутов О.Д. г.Ереван 1989г."
     AT(6%,16%)"ПРЕДСТАВЛЯЕТ ИГРУ:"
1240 FOR J%=10% TO 19%
1245 READ A%
1250 FOR I%=30% TO J% STEP -1%
1255 ? AT(I%,20%)CHR$(A%)" "
1260 FOR K%=1% TO 50%
1265 NEXT K%,I%,J%
1280 IF INKEY$="" THEN 1280 ELSE 25
```

Заключение рецензента

Программа Любутова О.Д. реализована по аналогии с известной игровой программой "Арканоид", работающей, например, на IBM PC.

При сокращении автором программы допущена ошибка при вводе кодов "стен". Хотя игра внешне работает (и очень хорошо работает!), добраться до девятого или даже до шестого уровня невозможно, так как на пятом уровне последние два "кирпича" не выбиваются, сколько бы игрок ни старался.

Высокая скорость перемещения графических объектов, достигнутая за счет написания отдельных подпрограмм в машинных кодах, случайная последовательность призов , достаточное количество внешних эффектов и звуковое сопровождение — все вместе обусловливает отличный игровой эффект программы, требующей хорошего внимания, железных нервов и вратарской реакции в отдельные игровые моменты. Игра доставит много приятных минут как любителям поиграть, так и любителям покопаться в чужих программах.

Жариков Л.Н.

О.Д.Любутов (г.Ереван)

"Вертолет"

Игра "Вертолет" представляет собой аналог игры "Хью-Кобра", реализованной на компьютере "Ямаха" (MSX-1). Из-за аппаратных различий БК-0010 и "Ямахи" игра несколько отличается от аналога.

Игра написана на языке высокого уровня БЕЙСИК с использованием вставок в машинных кодах. Занимает практически все программное ОЗУ (7,5 кБ).

Правила игры

Игрок (пилот) должен, управляя вертолетом с помощью стрелочных клавиш (вправо,влево,вверх,вниз), пройти через зону ПВО противника, состоящую из двух типов зенитных ракет и "летающих тарелок". Игрок (пилот) должен набрать наибольшее количество очков, подавляя наземые и воздушные объекты ПВО противника с помощью бортового оружия. В качестве бортового оружия используются бомбы (по наземным целям) и ракеты класса "воздух — воздух" (по воздушным целям).

Цикл игры состоит из семи этапов, упорядоченных по возрастанию сложности:

1 этап. ПВО противника использует только зенитные ракеты вертикального полета. Одновременно может быть задействована всего одна ракета.

2 этап. ПВО противника использует только зенитные ракеты наклонного полета. Одновременно находятся в полете четыре ракеты.

3 этап. ПВО противника использует только "летающие тарелки". Одновременно в полете четыре "тарелки".

4 этап. ПВО противника использует ракеты вертикального и наклонного полета. В полете пять ракет.

5 этап. ПВО противника использует ракеты вертикального взлета и "летающие тарелки". Одновременно в полете пять объектов.

6 этап. То же, что и в предыдущем этапе, только вместо вертикальных ракет — наклонного полета. Число объектов восемь.

7 этап. ПВО противника пускает в ход все резервы. В полете девять различных объектов.

Для прохождения каждого этапа игроку необходимо набрать определенное количество очков.

Игра развивает реакцию, умение рассредоточивать внимание на несколько объектов и ориентироваться в быстро меняющейся обстановке.

Рекомендации

Все комментарии, находящиеся в начале программы, следует исключить. Сначала набрать все строки с операторами DATA (1220 — 1360). Затем набрать следующую программу:

```
1 S!=0
2 FOR I=1 TO 213
3 READ A%
4 S!=S!+A%
5 NEXT I
6 PRINT S!
```

Запустить программу на выполнение. Если в результате появится число 798996, то все в порядке. Наберите DELETE-6 и продолжайте набор программы. В противном случает следует проверить правильность набора блока операторов DATA.

Описание работы программы

Ограничусь кратким описанием наиболее сложных мест программы. В программе реализованы две подпрограммы в машинных кодах. Одна подпрограмма (восьмеричные адреса 440 — 516) служит для циклического перемещения пейзажа под вертолетом. Другая (520 — 550) служит для перемещения графических объектов по экрану во время игры (вертолета, ракет, "тарелок"). Обе эти подпрограммы используют для своих целей третью подпрограмму (400 — 436) в качестве универсального модуля. Тексты этих подпрограмм находятся в следующих строках:

Универсальный модуль — строка 1330.

Подпрограмма перемещения пейзажа — строки 1340 — 1350.

Подпрограмма перемещения графических объектов — 1360.

В строках 520 — 530 осуществляется опрос клавиатуры и в соответствии с состояниями клавиш задается приращение положения вертолета на экране.

В строке 550 осуществляется проверка на выход вертолета за рабочую зону (она чуть меньше, чем экран).

В строках 600 — 610 имитируется звук работы двигателя вертолета. В достаточно большой подпрограмме 1700 — 2040 реализован опрос клавиатуры на нажатие пробела и организация стрельбы снарядами и бомбами.

В подпрограмме 2450 — 2770 организовано перемещение пейзажа и появление наземных ракетных установок.

И, наконец, для тех, кто решил разобраться в программе, назначение массивов и переменных:

Массив KS%(1%) — координаты снарядов (2 шт.)

Массив KB%(1%) — координаты бомб (2 шт.)

Массив KR%(3%) — координаты ракет наклонного полета (4 шт.)

Массив KI%(3%) — координаты летающих тарелок (4 шт.)

Массивы KO%(3%),KX%(3%) служат для обозначения наземных ракетных установок (на тягачах и в шахтах).

Переменные:

DM% — номер этапа в цикле

WW% — номер цикла

KV% — координата вертолета

DV% — приращение координаты вертолета

RV% — координата ракеты вертикального полета

```
10 'Игровая программа *Вертолет*
10 'Любутов О.Д.  г. Ереван  ул. Церетели
10 'д. 46  кв. 38  тел. 42-33-96
10 'Версия 1.1   1989 г.
10 DIM KS%(1%),KB%(1%),KR%(3%),KI%(3%),
   KO%(3%),KX%(3%)
20 DW%=1%
30 DEF USR0=&0440
40 DEF USR1=&0520
50 ?CHR$(&0224)CHR$(&0214)CHR$(&0214)
   CHR$(&0236)
60 PSET (0%,190%),3%
70 DRAW "S16C3E2R3E2R1F1D3R2U3E1F1D3R7E4F2R1
   E2R1D4R2U3F2D1R7U4;"
80 DRAW "C3R2F1E1R1F2E1F1E1D3R2U3E1F2D2R8D8L
   64;"
90 PAINT (1%,221%),2%,3%
100 COLOR 2%
110 ? AT(0%,1%);"СЧЕТ:"
120 ? AT(13%,22%);"=LOD="
130 FOR T%=1% TO 11%
140 GOSUB 1370
150 NEXT T%
160 FOR T%=&0400 TO &0550 STEP 2%
170 READ TT%
180 POKE T%,TT%
190 NEXT T%
200 GOSUB 1100
210 GOSUB 400
220 FOR R%=1% TO 5%
230 IF (R% MOD 2%)=0% THEN 250
240 GOSUB 520
250 ON DW% GOTO 280,340,310,270,300,330,260
260 GOSUB 2450
270 GOSUB 630
280 GOSUB 1440
290 GOTO 350
300 GOSUB 1440
310 GOSUB 2450
320 GOTO 350
330 GOSUB 2450
340 GOSUB 630
350 NEXT R%
360 IF HV%=0% THEN 210
370 GOSUB 1030
380 HV%=0%
390 GOTO 210
400 S%=S%+1%
410 IF S%<>21% THEN 500 ELSE S%=0%
420 POKE &0402,&040012
430 POKE &0406,&067671
440 POKE &0522,&06
450 A%=USR1(A%)
460 POKE &0402,&040022
470 POKE &0406,&067664
480 POKE &0522,&02
490 A%=USR1(A%)
500 A%=USR0(A%)
510 RETURN
520 DV%=0%
521 IF INP(&0177716,64%)<>0% THEN 530
522 IF INP(-78%,&0177)=&010 THEN DV%=-1%
523 IF INP(-78%,&0177)=&031 THEN DV%=1%
524 IF INP(-78%,&0177)=&032 THEN DV%=-128%
525 IF INP(-78%,&0177)=&033 THEN DV%=128%
530 KV%=KV%+DV%
540 GOTO 1690
550 IF KV%<&044700 OR KV%>&062350 OR (KV%
    MOD 64%)>40% THEN KV%=KV%-DV%
560 POKE &0402,&040001
570 POKE &0406,KV%
580 POKE &0522,&010
590 A%=USR1(A%)
600 POKE &0177716,64%
610 POKE &0177716,0%
620 RETURN
630 POKE &0522,&03
640 IF NN%=3% THEN 690
650 IF ABS(S1%-S%)<4% THEN 690 ELSE S1%=S%
660 NN%=NN%+1%
670 KO%(NN%)=-64%
680 KR%(NN%)=&066175-S%
690 I6%=-1%
700 I6%=I6%+1%
710 IF I6%>NN% THEN RETURN ELSE KR%(I6%)=
    KR%(I6%)+KO%(I6%)*4%-1%
720 IF KR%(I6%)<&045300 THEN KO%(I6%)=
    -KO%(I6%)
730 PE%=PEEK(KR%(I6%)+&0300)
740 POKE &0402,&040122-KO%(I6%)
```

```
750 POKE &0406,KRZ(I6Z)
760 AZ=USR1(AZ)
770 IF (KRZ(I6Z) MOD 64Z)<=0Z THEN GOSUB 830
780 IF PEZ=0Z OR PEZ=-17408Z THEN 700 ELSE
    XXZ=KRZ(I6Z)
790 GOSUB 1590
800 GOSUB 850
810 GOSUB 890
820 GOTO 700
830 XXZ=KRZ(I6Z)
840 GOSUB 1650
850 KRZ(I6Z)=KRZ(NNZ)
860 KOZ(I6Z)=KOZ(NNZ)
870 NNZ=NNZ-1Z
880 RETURN
890 IF ABS(KVZ-XXZ)<&02200 AND ABS(((KVZ+&03)
    MOD 64Z)-(XXZ MOD 64Z))<6Z THEN 970
900 FOR IZ=0Z TO 1Z
910 IF ABS((XXZ MOD 64Z)-(KBZ(IZ) MOD 64Z))>
    2Z OR ABS(KBZ(IZ)-XXZ)>&02100 THEN 950
920 XXZ=KBZ(IZ)
930 KBZ(IZ)=0Z
940 GOSUB 1590
950 NEXT IZ
960 RETURN
970 FOR JZ=1Z TO 3Z
980 FOR XXZ=KVZ+1Z TO KVZ+5Z STEP2Z
990 GOSUB 1590
1000 NEXT XXZ
1010 NEXT JZ
1020 OOZ=(40Z*WWZ+200Z)*(DWZ-1Z+7Z*WWZ)
1030 KSZ(1Z)=0Z
1040 KBZ(1Z)=0Z
1050 KBZ(0Z)=0Z
1060 KSZ(0Z)=0Z
1070 FOR IZ=2Z TO 16Z
1080 ? AT(0Z,IZ)SPC(32Z);
1090 NEXT IZ
1100 LINE (0Z,20Z)-(255Z,20Z),3Z
1110 ? AT(16Z,1Z);"ЭТАП:";DWZ+WWZ*7Z
1120 ? AT(6Z,1Z);OOZ;"  "
1130 KVZ=&054027
1140 RRZ=-1Z
1150 NNZ=-1Z
1160 RVZ=0Z
1170 FOR IZ=1Z TO 80Z
1180 GOSUB 400
1190 GOSUB 520
1200 NEXT IZ
1210 RETURN
1220 DATA &040202,&040702,&052074,&0327,
    &0327,&0174,&052500,&052400
1230 DATA &040204,&041740,&052525,&010000,
    &052400,&052524,&02005,&02005,&02004,
    &052524,&052520,&052520,&052500,&0401,
    &0404,&052520
1240 DATA &040206,&041706,&052525,&00,&01,
    &0125,&0404,&02004,&010004,&052525,
    &012525,&02525,&0525,&040100,&010100,
    &02525
1250 DATA &040312,&042012,&0140000,&030000,
    &06000,&06000,&0177700,&0177760,&0177760,
    &0177760,&0125250,&0170076,&06076,&06010,
    &0170040,&0125200
1260 DATA &040314,&042014,&01777,&01403,
    &01403,&01403,&01777,&01777,&01777,
    &01777,&0125252,&01700,&0146063,&0146063,
    &01700,&0125252
1270 DATA &040116,&042016,&0300,&01760,
    &07674,&037677,&01760,&01760,&01760,
    &07674,&037677,&036217,&025252,&0136017,
    &0136060,&020060,&04017,&01252
1280 DATA &040422,&041722,&0300,&01760,
    &01660,&07674,&037677,&01760,&01760,
    &01760,&01760,&07674,&037677,&036217
1290 DATA &040326,&042026,&010101,&012105,
    &012525,&012525,&02524,&0520,&0520,
    &02524,&012525,&012525,&012525,&02524,
    &0520,&0100
1300 DATA &040432,&041532,&014,&0124060,
    &0125200,&0125250,&0177752,&0125250,
    &0125200,&0124360,&060,&01777
1310 DATA &040434,&041534,&030000,&06052,
    &01252,&025252,&0125777,&025252,&01252,
    &07452,&06000,&0177700
1320 DATA &040242,&041742,&011001,&0100,
    &046014,&01440,&020104,&07343,&044620,
    &03564,&031642,&0107100,&03622,&011000,
    &020,&041001
1330 DATA &012700,&00,&012701,&00,&012702,
    &022,&0111011,&062700,&0100,&062701,
    &0100,&05302,&01402,&0137,&0414,&0203
1340 DATA &012737,&067600,&0402,&012737,
    &067677,&0406,&04337,&0400,&012704,&077,
    &012705,&067600
1350 DATA &0110537,&0406,&0105205,&0110537,
    &0402,&04337,&0400,&05304,&01402,&0137,
    &0470,&0207
1360 DATA &012704,&00,&04337,&0400,&05237,
    &0402,&05237,&0406,&05304,&01402,&0137,
    &0524,&0207
1370 READ IIZ
1380 READ JJZ
1390 FOR IZ=IIZ TO JJZ STEP 64Z
1400 READ KZ
1410 POKE IZ,KZ
1420 NEXT IZ
1430 RETURN
1440 POKE &0522,&02
1450 IF RVZ<>0Z THEN 1480
1460 QZ=1Z-QZ
1470 RVZ=&066135-3Z+QZ*21Z
1480 RVZ=RVZ-256Z
1490 POKE &0402,&040222
1500 POKE &0406,RVZ
1510 PEZ=PEEK(RVZ)
1520 AZ=USR1(AZ)
1530 IF PEZ=0Z AND RVZ>&046000 THEN RETURN
1540 XXZ=RVZ
1550 GOSUB 1590
1560 RVZ=0Z
1570 IF XXZ>&046000 THEN GOSUB 890
1580 RETURN
1590 POKE &0402,&040042
1600 POKE &0406,XXZ
1610 AZ=USR1(AZ)
1620 BEEP
1630 BEEP
```

```
1640 BEEP
1650 POKE &0402,&040066
1660 POKE &0406,XX%
1670 A%=USR1(A%)
1680 RETURN
1690 E1%=E%
1700 E$=INKEY$
1710 II%=-1%
1720 II%=II%+1%
1730 IF II%>1% THEN 550
1740 IF E$<>" " OR E1%<>0% THEN 1770
1750 IF KS%(II%)=0% THEN KS%(II%)=KV%+582%
1760 IF KB%(II%)=0% THEN KB%(II%)=KV%+&01404
1770 IF KS%(II%)=0% THEN 1860
1780 POKE KS%(II%),0%
1790 KS%(II%)=KS%(II%)+2%
1800 IF PEEK(KS%(II%))=0% THEN 1850
1810 XX%=KS%(II%)
1820 GOSUB 2150
1830 KS%(II%)=0%
1840 GOTO 1860
1850 IF (KS%(II%) MOD 64%)<=4% THEN KS%(II%)=
     0% ELSE POKE KS%(II%),&0125252
1860 IF KB%(II%)=0% THEN 1980
1870 KB%(II%)=KB%(II%)+&0300
1880 IF PEEK(KB%(II%)+&02000)=0% AND KB%(II%)
     <&065300 THEN 1940
1890 XX%=KB%(II%)
1900 GOSUB 1590
1910 IF KB%(II%)<&065300 THEN GOSUB 2150
     ELSE GOSUB 1990
1920 KB%(II%)=0%
1930 GOTO 1980
1940 POKE &0402,&040026
1950 POKE &0522,&02
1960 POKE &0406,KB%(II%)
1970 A%=USR1(A%)
1980 GOTO 1720
1990 IE%=0%
2000 IF ABS(((&067672-S%-21%*IE%) MOD 64%)-
     (XX% MOD 64%))<3% THEN 2050
2010 IF ABS(((&067663-S%-21%*IE%) MOD 64%)-
     (XX% MOD 64%))<2% THEN 2100
2020 IE%=IE%+1%
2030 IF IE%<=2% THEN 2000
2040 RETURN
2050 POKE &0522,&06
2060 XX%=&067570-S%-21%*IE%
2070 GOSUB 1590
2080 OO%=OO%+8%
2090 GOTO 2360
2100 POKE &0522,&02
2110 XX%=&067563-S%-21%*IE%
2120 GOSUB 1590
2130 OO%=OO%+17%
2140 GOTO 2360
2150 I6%=0%
2160 IF I6%>RR% THEN 2180
2170 IF ABS(KT%(I6%)-XX%)<&02200 AND ABS(((
     KT%(I6%)+2%) MOD 64%)-(XX% MOD 64%))<6%
     THEN 2240
2180 IF I6%>NN% THEN 2220
2190 IF ABS(KR%(I6%)-XX%)<&02200 AND ABS((
     KR%(I6%) MOD 64%)-(XX% MOD 64%))<2%
     THEN 2280
2200 I6%=I6%+1%
2210 IF I6%<=3% THEN 2160
2220 IF ABS(RV%-XX%)<&02200 AND ABS((RV% MOD
     64%)-(XX% MOD 64%))<2% THEN 2310
2230 RETURN
2240 XX%=KT%(I6%)
2250 POKE &0522,&04
2260 GOSUB 2740
2270 GOTO 2340
2280 XX%=KR%(I6%)
2290 GOSUB 850
2300 GOTO 2330
2310 XX%=RV%
2320 RV%=0%
2330 POKE &0522,&02
2340 GOSUB 1590
2350 OO%=OO%+3%
2360 ? AT(6%,1%);OO%
2370 IF OO%<(40%*WW%+200%)*(DW%+WW%*7%)
     THEN RETURN
2380 DW%=DW%+1%
2390 IF DW%<8% THEN 2430
2400 DW%=1%
2410 WW%=WW%+1%
2420 IF WW%<9% THEN 2430 ELSE WW%=1%
2430 HV%=5%
2440 RETURN
2450 POKE &0522,&05
2460 FF%=FF%+1%
2470 IF RR%=3% OR FF%<27% THEN 2520
2480 FF%=0%
2490 RR%=RR%+1%
2500 KT%(RR%)=&045573+INT(RND(1%)*70%)*64%
2510 KX%(RR%)=SGN(RND(1%)-0.5!)*64%
2520 I6%=-1%
2530 IF RR%<0% THEN RETURN
2540 I6%=I6%+1%
2550 IF I6%>RR% THEN RETURN
2560 KT%(I6%)=KT%(I6%)+KX%(I6%)*4%-1%
2570 IF KT%(I6%)>&045200 AND KT%(I6%)<&062400
     THEN 2640
2580 KX%(I6%)=-KX%(I6%)
2590 IF (KT%(I6%) MOD 64%)>1% THEN 2640
2600 XX%=KT%(I6%)
2610 GOSUB 2740
2620 GOSUB 1650
2630 GOTO 2540
2640 PE%=PEEK(KT%(I6%)+256%)
2650 POKE &0402,&040132-KX%(I6%)
2660 POKE &0406,KT%(I6%)
2670 A%=USR1(A%)
2680 IF PE%=-22% OR PE%=-5632% OR PE%=0%
     THEN 2540
2690 XX%=KT%(I6%)
2700 GOSUB 1590
2710 GOSUB 2740
2720 GOSUB 890
2730 GOTO 2540
2740 KT%(I6%)=KT%(RR%)
2750 KX%(I6%)=KX%(RR%)
2760 RR%=RR%-1%
2770 RETURN
```

К.Л.Погосова (г.Кутаиси)

Игра в крестики-нолики

Я предлагаю владельцам КУВТ "КОРВЕТ" программу всем хорошо известной игры в крестики-нолики, которую я назвала так: X-O. BAS.

Программа предназначена для двух игроков, которые поочередно ходят после появления вопросительного знака. Игрок должен набрать одно из чисел, указанных в клетках (1...9), и нажать на клавишу. Сообщение о выигрыше появляется на стороне игрока. Возможен ничейный результат.

Теперь немного о программе.

Строки 10 — 110 — название игры

- 130 — 260 рисуют поле 3x3

- 270 — 340 позволяют игрокам сделать свои ходы, которые определяются функцией RND

- 360 — 440 определяют по значению N (1...9), где надо нарисовать крестик или нолик. А что именно рисовать — строками 460-530.

В строках 350, 450, 520 — 620 происходит поверка выигрыша или ничейного варианта.

Замечу, что программа хорошо работает на РМП, а на РМУ дает синтаксическую ошибку в 270-й строке. Вам придется просто стереть эту строку в программе для РМУ.

```
10 REM--- КРЕСТИКИ-НОЛИКИ-------1989г.-----Погосова К.Л.-----
20 CLS:PCLS
30 PRINT STRING$(50,"*")
40 PRINT "*"TAB(49)"*"
50 PRINT"*"TAB(49)"*"
60 PRINT "*" SPC(16)"КРЕСТИКИ-НОЛИКИ" SPC(17) "*"
70 PRINT"*"TAB(49)"*"
80 PRINT"*"TAB(49)"*"
90 PRINT STRING$(50,"*")
100 PRINT:PRINT:PRINT
110 PRINT:PRINT:PRINT:PRINT"нажмите на любой клавиш "
120 IF INKEY$= "" THEN 120
130 CLS
140 PRINT SPC(15) STRING$(31,"~")
150 PRINT SPC(15)"¦1         ¦2         ¦3         ¦"
160 PRINT SPC(15)"¦          ¦          ¦          ¦"
170 PRINT SPC(15)"¦          ¦          ¦          ¦"
180 PRINT SPC(15) STRING$(31,"~")
190 PRINT SPC(15)"¦4         ¦5         ¦6         ¦"
200 PRINT SPC(15)"¦          ¦          ¦          ¦"
210 PRINT SPC(15)"¦          ¦          ¦          ¦
220 PRINT SPC(15)STRING$(31,"~")
230 PRINT SPC(15)"¦7         ¦8         ¦9         ¦"
240 PRINT SPC(15)"¦          ¦          ¦          ¦"
250 PRINTSPC(15)"¦          ¦          ¦          ¦"
260 PRINT SPC(15) STRING$(31,"~")
270 RANDOMIZE 1
280 U=1
290 A=INT(RND(1)*2+1)
300 T1=1:T2=1:IF A=1 THEN P=2 ELSE P=1
310 LOCATE 1,T1:PRINT A;"  нолики"
320 LOCATE 50,T2:PRINT P;"  крестики"
330 T1=T1+1:T2=T2+1
340 IF P=2 THEN LOCATE 1,T1:INPUT N:X(R)=N:A$(N)="O":A$="НОЛИКИ" ELSE
LOCATE 50,T2 :INPUT N:X(R)=N:A$(N)="X":A$="КРЕСТИКИ"
```

```
350 U=U+1
360 IF N=1 THEN X=160:Y=40:GOTO 460
370 IF N=2 THEN X=245:Y=40:GOTO 460
380 IF N=3 THEN X=325:Y=40:GOTO 460
390 IF N=4 THEN X=160:Y=100:GOTO 460
400 IF N=5 THEN X=245:Y=100:GOTO 460
410 IF N=6 THEN X=325:Y=100:GOTO 460
420 IF N=7 THEN X=160:Y=162:GOTO 460
430 IF N=8 THEN X=245:Y=162:GOTO 460
440 IF N=9 THEN X=325:Y=162:GOTO 460
450  LOCATE 25,14:PRINT"* ** * НИЧЬЯ * ** *":FOR I=1 TO 5:BEEP:NEXT I:
GOTO 660
460 IF A$="НОЛИКИ" THEN 510 ELSE 470
470 REM
        КРЕСТИКИ
480 LINE(X-10,Y-15)-(X+10,Y+15):LINE(X-9,Y-15)-(X+11,Y+15)
490 LINE(X-10,Y+15)-(X+10,Y-15):LINE(X-9,Y+15)-(X+11,Y-15):B$="X"
500 T2=T2+1:SWAP A,P:GOTO 540
510 REM НОЛИКИ
520 CIRCLE(X,Y),19,,,,.7:CIRCLE(X,Y),20,,,,.7:B$="O"
530 T1=T1+1:SWAP A,P:GOTO 540
540 FOR I=1 TO 7 STEP 3
550 IF A$(I)=B$ AND A$(I+1)=B$ AND A$(I+2)=B$ THEN BEEP:GOTO 640
560 NEXT I
570 FOR I=1 TO 3
580 IF A$(I)=B$ AND A$(I+3)=B$ AND A$(I+6)=B$ THEN BEEP:GOTO 640
590 NEXT I
600 IF A$(1)=B$ AND A$(5)=B$ AND A$(9)=B$ THEN BEEP:GOTO640
610 IF A$(3)=B$ AND A$(5)=B$ AND A$(7)=B$ THEN BEEP:GOTO 640
620 IF U=10 THEN 450
630 GOTO 340
640 IF B$="O" THEN LOCATE 1,T1 ELSE LOCATE 50,T2
650 PRINT "ВЫ ПОБЕДИЛИ":BEEP:BEEP:BEEP:BEEP
660 END
```


"ЦЕНА ЗНАНИЙ В НАШЕМ ОБЩЕСТВЕ ПАЛА В ПОСЛЕДНИЕ ГОДЫ ЧРЕЗВЫЧАЙНО НИЗКО".

Из беседы с Д.Граниным ("Известия", 1991, 23 июля)

ПОЗВОЛЬТЕ НЕ СОГЛАСИТЬСЯ!

Цена на знания, предоставляемые нашим издательством, повышается из года в год.

В 1992 году один выпуск серии "Вычислительная техника и ее применение" будет стоить 90 коп.

Стоимость годовой подписки — 10 руб. 80 коп.

Р.А.Сузи
(г.Петрозаводск)

Игра "ЖИЗНЬ"

Предлагаемая программа выдает два кадра за секунду, а для ее набора и ассемблирования требуется не более 20 минут.

Инструкция

Введите программу, отассемблируйте ее и перенесеите в адреса: 0100 — 01CE.

Запуск

C100

Теперь к вашим услугам экранный редактор "СИСТЕМНОГО МОНИТОРА": СТР — стирание с экрана, ПС — поможет сдвинуть рисунок вверх, работают клавиши управления курсором.

На чистом поле экрана можно расставить колонки клеток: в нужные места экрана ставятся символы, двоичный код содержит 1 в 0-м двоичном разряде. Это: Ш,О и др. Другие символы означают отсутствие клетки. Это: :

Для запуска на эволюцию нужно нажать ВК. На экране будет меняться картинка.

Если надо остановить эволюцию и изменить колонку клеток, нажимается клавиша "стрелка влево", и вновь работает экранный редактор.

Об игре

Ее придумал американский математик Дж.Х.Конуэй.

Открыто много различных комбинаций клеток, и они хорошо описаны.

Могу предложить книгу М.Гарднера "Крестики-нолики" — там в трех главах описана эта игра.

А для любителей совершенствовать программы можно предложить:

- развернуть TAKE и TAKEP на места вызова;
- сделать счетчик поколений;
- "заторить" экран, т.е. "склеить" низ и верх; левый край и правый — теперь существа будут жить в замкнутом пространстве;
- поэкспериментировать с "генетическими" законами.

ТЕКСТ ПРОГРАММЫ

```
"МИКРОША", 580, Р.СУЗИ, 1990,
ПЕТРОЗАВОДСК
    ORG 100H; "ЖИЗНЬ"
    BEGIN: MVI C, 31
    GETCYCL: CALL PUTCHAR
    NEWPIC: CALL GETCHAR
    CPI 13
    MOV C, A
    JNZ GETCYCL
    AGAIN: LXI H, BUF 0
    MVI A, 128
    CLCYCL: MOV M, H
    INR L
    DCR A
    JNZ CLCYCL
    STA BUFNOM
    MVI C,1
    STRCYCL: CALL PUTBUF
    CALL GETVAS
    LXI D, 78
    POSCYCL: INX D
    MOV A,M
    ANI 1
    ADD A
    ADD A
    ADD A
    CALL TAKEPL
    DCX H
    CALL TAKE
    DCX H
    CALL TAKE
    LXI D 0FF B2H; =-78
    CALL TAKEPL
    CALL TAKEPL
    INX H
```

```
CALL TAKE
INX H
CALL TAKE
LXI D, 78
CALL TAKEPL
SUI 3
JZ SET; 3 соседа — рождение клетки
SUI 7
JZ SET; (3+7)-8=2 соседа-сохранение
DCR A
JZ SET; (3+7+1)-8=3 соседа-сохранение
MVI A, NOTCELL
JMP ENDIF
SET: MVI A, CELL
END IF: DB 32H; = STA
BUFPOINT: DS 2
LDA BUFPOINT
INR A
STA BUFPOINT
DCR B
JNZ POSCYCL
INR C
MOV A, C
CPI 26
JM STRCYCL
CALL PUTBUF
INR C
CALL PUTBUF
CALL GETKEY
CPI 8
JNZ AGAIN
JMP NEWPIC
TAKEPL: DAD D
TAKE: STA TMP
MOV A, M
ANI 1
DB 0C6H ; = ADI
TMP: DS 1
RET
PUTBUF: DB 3EH ; = MVI A,...
BUFNOM: DS 1
XRI 1
STA BUFNOM
LXI H, BUF0
JZ ENDIF1
MVI L, BUF1
ENDIF1: SHLD BUFPOINT
DCR C
DCR C
PUSH H
CALL GETVAS
POP D
PUTCYCL: LDAX D
MOV M, A
INX H
INR E
DCR B
JNZ PUTCYCL
INR C
INR C
RET
GETVAS: MVI H, 0
MOV D, H
MOV E, C
MOV L, C
DAD H; HL:=C*78 ↓
DAD H
DAD H
DAD D
DAD H
DAD D
DAD H; HL=C*78 ↑
LXI D, 76D0H+78+78+7+1
DAD D
MVI B, 64
RET
CELL: EQU 17H ; = '■'
NOTCELL; EQU '•' ; "точка"
PUTCHAR: EQU 0F809H
GETCHAR: EQU 0F803H
GETKEY: EQU 0FEEAH
BUF0: EQU 0
BUF1: EQU BUF0+64
END
```

А.Бугайнов
(г.Усть-Каменогорск)

"Быки и коровы"

Предлагаю свою версию программы логической игры "Быки и коровы". В продаже эта игра была под названием "Узнай код". Она представляла собой пластмассовое игровое поле и кучу фишек разного цвета: игровых шести цветов примерно по 12 каждого цвета, черных и белых для информации о количестве быков и коров.

Суть игры в следующем: из шестицветного набора фишек один игрок загадывает комбинацию из четырех фишек разного цвета, размещает их по своему усмотрению и закрывает экраном. Другой игрок делает ход, выставляя четыре фишки разного цвета на игровое поле. Тот, кто загадывает, дает информацию о наличии быков и коров. Если фишка совпала и по цвету и по месту нахождения — это бык; если цвет фишки присутствует в загаданной комбинации, но фишка находится не на своем месте — это корова. Информация дается только о количестве быков и коров в каждом ходе.

Делая ходы и используя получаемую информацию, второй игрок пытается разгадать загаданную комбинацию, т.е. определить четырех быков. Если такое удается сделать не более чем за 5 — 6 ходов, то это свидетельствует о хорошем уровне логического мышления.

В моей программе роль загадывающего исполняет компьютер. Игровое поле занимает левую половину экрана. В правой половине размещается краткая инструкция, как делать ходы. В нижней части правой половины, под инструкцией, размещены шесть фишек разного цвета.

Все цвета пронумерованы: 1 — желтый, 2 — фиолетовый, 3 — голубой, 4 — зеленый, 5 — красный, 6 — синий. Формирование фишки определенного цвета происходит в подпрограмме ФИШКА (строки 1400 — 1480). Сама фишка — это ряд концентрических окружностей радиусом от одного до восьми с шагом единица.

Дополнительные три цвета получаются: желтый — чередованием красных и зеленых окружностей, фиолетовый — красных и синих, голубой — синих и зеленых. Загадывание, или, иначе сказать, кодирование четырехцветной комбинации фишек, происходит в строках 120 — 230. Закодированная комбинация хранится до конца игры в памяти машины в массиве M% (4).

При запуске программы на игровом поле, заключенном в красную рамку, появляется цифра 1 (номер хода) и четыре пронумерованных (от 1 до 4) кружка; ниже — строчка НАБОР:? и курсор; это по команде INPUT (строка 850) компьютер ждет выдачи фишек. Фишка выставляется набором через запятую номера позиции и номера цвета и нажатием клавиши <ввод>. Эта операция повторяется, пока не будет набрана нужная одноходовая комбинация четырех фишек.

Если какая-либо уже выставленная фишка не устраивает играющего, ее можно заменить, проделав вышеуказанную операцию набора; но сделать это можно только до фиксации хода. Набранная комбинация размещается в массиве D% (4).

Этот массив непостоянен; значения переменных в нем хранятся только до сравнения с загаданной комбинацией, т.е. до сравнения с массивом M% (4), во время которого происходит подсчет быков и коров (строки 1090 — 1190) и выдается письменная информация о количестве быков и коров (например, 063 или 261к или по-другому), которая размещается рядом с набранной комбинацией и хранится на экране на весь цикл игры. Со следующим ходом значения переменных массива D% (4) могут изменяться. Отдельные ходы фиксируются набором через запятую двух нулей и нажатием клавиши <ввод>. Программой предусмотрена реакция на ошибки (строки 870 — 920 и 970 — 1070).

Логическая игра
"Быки и коровы"
Автор программы:
Бугайнов А.В.

```
10  REM
20  REM
30  REM
40  REM
50  REM
60  REM
70  REM
80  REM
90  REM
100 REM начальная установка
110 DIM M% (4), D% (4)
120 REM кодирование
130 FOR J%=1 TO 4
140 N%  =INT (RND (1) * 6) +1
150 K% = 0
160 K% = K% + 1
170 IF  K% = J% THEN 200
180 IF  M% (K%) = N% THEN 140
190 GOTO 160
200 M% (K%) = N%
210 D% (J%) = 0
220 NEXT
230 ' конец кодирования
300 REM инструкция, игровое поле
310 CLS
320 ?
330  ?  TAB  (17)  CHR$(156);" Быки и
коровы"; CHR$ (156)
340 ? TAB  (19) "Из шестицвет-        ";
350 ? TAB  (17) "ного набора за-      ";
360 ? TAB  (17) "гадана 4-х цвет-     ";
370 ? TAB  (17)  "ная комбинация      ";
380 ? TAB  (17)  "фишек. На разга-    ";
390 ? TAB  (17)  "дывание дается      "
400 ? TAB  (17)  "максимум 10 хо-     ";
410 ? TAB  (17) "дов. 5-6 означа-     ";
420 ? TAB  (17)  "ют хороший уро-     ";
430 ? TAB  (17)  "вень логическо-     ";
440 ? TAB  (17)  "го мышления. На-    ";
450 ? TAB  (17) "бор хода: / пози-    ";
460 ? TAB  (17)  "ция / , / цвет / кла-";
470 ? TAB  (17) "виша <ввод> / По-    ";
480 ? TAB  (17) "вторить. Набрав"
490 ? TAB  (17)   "нужную комбина-  ";
500 ? TAB  (17)  "цию, фиксируйте"
510 ? TAB (17)   "ход вводом 0, 0       "
520 ? AT (1, 22) CHR$ (1 4 7); "четыре
быка"; CHR$ (145);"  1  2  3  4  5  6"
530 Y% = 225
540 FOR N%=1  TO 6
550 X%=N% * 24 + 102
560 GOSUB 1400
570 NEXT
580 LINE (0, 0)  —  (128, 210), 2, B
590 LINE (0, 210)  —  (104, 239), 2, B
700 REM полный игровой цикл
710 FOR I% = 1 TO  10
720 Y% = I% * 20  —  5
730 REM цикл одного хода
740 IF I% = 10 TNEN 770
750 ? AT (1, I% * 2  —  1) CHR$ (I% +48) +"
1  2  3  4 "
760 GOTO 800
770 LINE (0, 210)  —  (103, 210), 0
780 ? AT (1, 22)"
790 ? AT (0, 19)" 10 1 2 3 4 "
800 FOR X%=28 TO 76 STEP 16
810 CIRCLE (X%, Y%), 8,,,, 1.2
820 NEXT
830 ? AT (1, I% * 2 + 1)"              "
840 ? AT (1, I% * 2 + 1)" НАБОР:";
850 INPUT J%, N%
860 IF J% = 0  AND N% = 0  THEN 970
870 IF J% > 0 AND J% < 5 AND N% > 0
AND N% < 7 THEN 930
880 ? AT (1, I% * 2 + 1) "ОШИБКА!!!  "
890 ? STRING$ (15, 7)
900 FOR Z% = 1 TO  6000
910 NEXT
920 GOTO 830
930 D% (J%)=N%
940 X%=J% * 16 + 12
950 GOSUD 1400
960 GOTO 830
970  REM  ПРОВЕРКА  НА  КОМП-
ЛЕКТ
980 C%=0
990 FOR J%=1 TO 4
1000 IF D% (J%) = 0 THEN C%=C%+1
1010 NEXT
1020 IF C%=0 THEN 1080
1030  ?  AT  (1,  I%  *  2  +  1)  "НЕКОМП-
ЛЕКТ-" +CHR$  (C% + 48)
1040 ? STRING$ (15, 7)
1050 FOR Z%=1 TO 6000
1060 NEXT
1070 GOTO 830
1080 ? AT (1, I% * 2 + 1)"              "
1090  REM  ПОДСЧЕТ  БЫКОВ  И  КО-
РОВ
1100 Z% = 0
1110 FOR J%=1 TO 4
1120 FOR K%=1 TO 4
1130  IF  D%  (J%)  <>  M%  (K%)  THEN
1150
1140   IF   J%=K%   THEN   Z%=Z%+1
ELSE
C% = C% + 1
1150 NEXT
1160 D% (J%) = 0
1170 NEXT
1180 IF Z% = 4 THEN 1240
1190 ? AT (11, I% * 2  —  1) CHR$(Z%
+ 48) + "б" + CHR$ (C% + 48) + "к"
```

```
1200 NEXT
1210 ' игровой цикл завершен
1220 REM ОКОНЧАНИЕ ИГРЫ
1230 GOTO 1250
1240 ? AT (11, I% * 2 — 1) "БЫКИ"
1250 LINE (0, 210) — (103, 210), 2
1260 ? AT (1, 22)"      "
1270 Y%=225
1280 FOR K%=1 TO 4
1290 X%=K% * 16 + 12
1300 N%=M%(K%)
1310 GOSUB 1400
1320 NEXT
1330 ? AT (0, 0)
1340 END
1400 REM ПОДПРОГРАММА <ФИШ-
KA>
1410 FOR R%=1 TO 8
1420 IF N% < 4 THEN 1450
1430 C%=N%-3
1440 GOTO 1470
1450 Z%=R% MOD 2
1460 IF N%=3 THEN C% =Z%* (Z% + 1) +1
ELSE C%=Z%+N%
1470 GIRCLE (X%, Y%), R%, C%,,,
1 . 2
1480 NEXT
1490 RETURN
```

Независимо от того, за сколько ходов будет найдено 4 быка, или вообще они не будут найдены за 10 ходов (а весь цикл игры рассчитан на 10 ходов), в нижней части левой половины в отдельной рамке (там, где написаны синим цветом слова ЧЕТЫРЕ БЫКА) появится закодированная комбинация (строки 1220 — 1320).

На этом игра заканчивается. Для повторения игры следует перезапустить программу командой RUN либо нажатием ключевой клавиши [5] или [0] на альтернативном регистре АР2. По количеству сделанных ходов играющий может судить о своем уровне логического мышления. Более 6 ходов — это уже посредственный уровень.

С точки зрения профессионального программиста программа возможно имеет свои недостатки. Например, все переменные у меня обозначены, как целочисленные, но я сознательно сделал это; тем более что в подпрограмме ФИШКА есть переменные, которые "работают" с модульным делением.

Можно было бы также вывести в отдельную подпрограмму повторяющиеся строки 890 — 910 и 1040 — 1060, но я не нашел в этом выгоды, так как и в том и в другом случае получается шесть строк. А в общем программа отлажена и работоспособна. Листинг программы списан с экрана и проверен на три раза. Каждая клеточка соответствует одному знакоместу (в режиме 32).

Считаю необходимым сделать одну оговорку; дело в том, что программы я пишу для своего компьютера. а у него "входной" цвет зеленый, так как я поменял местами распайку контактов 3 и 5 на одном конце кабеля ЦТВ; поэтому в моих программах код 145 соответствует зеленому цвету, код 146 — красному. И также меняются цветовые аргументы: по умолчанию или единица соответствует зеленому цвету, двойка — красному. Тем, кто захочет использовать мою программу, если у их компьютеров "входной" цвет остается первоначально красным, следует внести изменения в программу. В начале программы вставить строку [COLOR 2,0] (например, под номером 105), тогда по умолчанию цвет будет зеленым; в строке 520 код CHR (145) следует на код CHR (146); в строках 580, 590 и 1250 следует заменить цветовой аргумент [2] на [1]. Также напоминаю о необходимости соблюдать все пробелы в кавычках.

А.В.Милюков (г.Загорск)

Все о ТЕТРИСЕ

Нижеследующее адресовано тем, кто любит игры по принципу "любишь кататься — люби и саночки возить". Завод "Нуклон", изготовивший в 1990 году мой БК, сопроводил эту модель двумя кассетами с играми. Не секрет, что многие из них или примитивны, или морально устарели. Однако это лучше, чем вообще ничего. Для тех, кто не против попытаться вдохнуть новую жизнь в старые игры, — несколько советов по поводу тетриса. К сожалению, я не могу однозначно сказать, о какой из версий пойдет речь, так как на данный момент их множество. Начальный адрес — 1000, длина — 17000. Поставляется заводом "Нуклон" в пакете игровых программ. При запуске появляется заставка — большие буквы ТЕТРИС на фоне экрана, заполненного строчкой =c11=. В остальном игра особенностей не имеет.

При детальном анализе листинга обнаружено, что кубики, которыми составлены падающие фигуры, есть не что иное, как инверсное знакоместо определенного цвета с напечатанным квадратом цвета фона внутри. Код "■" равен 177_8, и имеется ячейка с адресом 14344, которая содержит код 77577. Нетрудно убедиться, что такая запись соответствует двум одинаковым байтам для символа "■". Чтобы изменить вид кубиков, из которых составлены фигуры, нужно заменить код в этой ячейке. Например, 20040 (два пробела) или 25052 (две звездочки). Желающие могут самостоятельно составить нужные коды. Занести их необходимо до запуска игры, например в ТМС A=14344.

Помимо указанных изменений, легко модернизировать наскучившую победную трель Тетриса. Соответствующая подпрограмма начинается с адреса 1426. В R1 помещается 1000 (длительность), а в R0 — "период" звуковых колебаний, после чего идет обращение к подпрограмме 11156, которая "выдает" звук. Последовательность "нот" определяется содержимым ячеек 1434, 1444, 1454, 1464, ... 1524. Их содержимое также необходимо изменять (при желании) до запуска игры.

Для тех, кто неуютно чувствует себя при работе в машинных кодах, написана Репка. Прообразом ее послужил тот же Тетрис, однако концепция игры другая. Есть определенное сходство и с Питоном, множество версий которого уже стали хрестоматийными.. Цель — засадить поле — "стакан" репкой, не раздавив при этом ни одной и не попав в "лужи нефти". В игре практически не используются цветовые возможности БК, и графика почти полностью табличная (псевдо-). Это позволяет легче вникнуть в алгоритм тому, кто хочет что-либо усовершенствовать.

Прилагаемая распечатка требует комментариев ввиду того, что знакогенератор ТПУ не совпадает с знакогенератором ПЭВМ.

Массив T% (22,25) является копией игрового поля на экране. T%=3 соответствует пятну нефти, при попадании в любое из них игра заканчивается. T%=0 соответствует незасаженному полю, T%=1 соответствует "растоптанной" репке, на месте которой больше ничего не растет.

Игрок должен ответить на вопрос, впервые он играет или нет. Если нет, то выбрать скорость, иначе — прочесть правила игры. Эти действия выполняются в строках 30, 40. В правилах игры использован символ "♥", означающий репку (АР2+Б), "$" — затоптанная репка и — "#" нефть. Для того, чтобы иг-

рок прочитал, введена задержка: п/п 1140. После выбора скорости — строка 320 — происходит очистка всех переменных и игрового массива, проверка корректности выбора скорости и рисуется игровое поле. Для уменьшения занимаемого места в памяти использована символьная переменная Y$="||↓↓...↓↓||",которая формирует на экране рамку (AP2+"O"), внутри которой "трава" (AP2+"M"). Одновременно в правой части экрана появляется комментарий о том, что играющий должен избегать нефтяных пятен. Сделано это умышленно, но для опытного игрока теряет смысл.

В произвольном месте поля появляется первая репка, и если играющий не нажмет ни одну из клавиш управления курсором, она будет затоптана. Перед каждым ходом проверяется, что содержится в этом месте экрана. Происходит ветвление; п/п 1020 — если вы попали в нефть, и через определенное число ходов, завершившихся посадкой репки, — подпрограмма козни. Ее начало — 910 строка. В некоторые моменты времени — для игрока это неожиданность — появляются "россыпи" нефтяных пятен, каждое из которых прерывает ход игры. При этом расходятся круги в месте попадания в лужу, проходит пауза и игроку выводятся количество набранных очков, строка 1090 и запрос продолжения игры. Если выбранная скорость не нравится игроку, он в любой момент может прервать игру нажатием клавиши "И", код 85_{10}, и задать новую.

Число набранных очков пропорционально скорости в том виде, как ее вводит игрок, т.е. для 6-й всегда меньше, чем для 1-й. Количество луж также зависит от скорости игры.

Эта игра не претендует на роль постоянного спутника досуга, она скорее является основой, на которой можно создать нечто более интересное. Например, вместо массива в тех же целях использовать ОЗУ экрана и при помощи функции PEEK определять, что находится в данной позиции экрана. Возможно введение каких-либо иных игровых ситуаций или использование спрайтов для более реального изображения элементов игры. В исходном, почти "текстовом" игровом поле заинтересованы те, кто работает, например, на БЕЙСИКе ДВК, который является подмножеством рассматриваемой версии.

Хотелось бы надеяться, что все вышеизложенное представит определенный интерес для читателей рубрики "БК за рога". Предполагаю обрести таким образом многих единомышленников. Кроме того, по возможности прошу сообщить координаты организаций, занятых программным обеспечением БК, заинтересованных в контакте с пользователями, или любителей-программистов того же направления.

```
10 CLS
20 DIM T%(22,25)
30 INPUT "wY IGRAETE WPERWYE";LØ
40 IF LØ="N" THEN 300
50 ? "mi{ka u`bit repku i savat by"
60 ? "ee ♡ wezde,no on neuku`v i"
70 ? "top~et to,~to posadit 'Ø'i
   tam  uve ni~ego ne rastet"
80 GOSUB 1140
90 ? "***** prawila igry ****"
100 ? "*                    *"
110 ? "* wy legko pomovete   *"
120 ? "* mi{ke savatx repku: *"
130 ? "*                    *"
140 ? "* ↓-swerhu wniz      *"
150 ? "*                    *"
160 ? "* ←-wlewo            *"
170 ? "*                    *"
180 ? "* →-wprawo           *"
190 ? "*                    *"
200 ? "* ↑-sledu`Jaa repka  *"
210 ? "*                    *"
220 ? "*  poqwitsa w werhnej *"
230 ? "*                    *"
240 ? "*  qradke.spe{ite za- *"
250 ? "*                    *"
260 ? "*  datx kurs,ina~e 'Ø'*"
270 ? "*                    *"
280 ? "**********************"
```

```
290 INPUT "prowitali";LQ
300 ? " wybirajte skorostx
    posewa 1-6  w igre movno ee
    menatx,navaw 'U' latinsk
310 CLEAR
320 INPUT "wybrali";DL
330 IF DL<1 OR DL>6 THEN 300
340 YQ="      #++++++++++++++++#"
350 CLS
360 ? YQ+"AWTOR"
370 ? YQ+"PEPKI"
380 ? YQ+"aL-DR",YQ+"MILQKOW",
    YQ+" ZABYL"
390 ? YQ+"SKAZATX",YQ+"O KOZNQH"
400 ? YQ+"wragow "
410 ? YQ,YQ+"KOTORYE"
420 ? YQ+"NALILI",YQ+"KOE-GDE"
430 ? YQ+"NEFTX",YQ+"I TAM"
440 ? YQ+"  BY",YQ+"UTONETE",
    YQ+"S REPOJ"
450 FOR I=1TO5
460 ? YQ
470 NEXT I
480 ? "      <=================]"
490 X=10
500 Y=INT(18*RND(DL))
510 IF T%(X,Y)=3 THEN 1020
520 IF T%(X,Y)=0 THEN 540
530 GOTO 560
540 ? AT(X,Y);"♡"
550 G=G+1
560 IF T%(X,Y)=1 THEN 580
570 GOTO 600
580 ? AT(X,Y);"Q"
590 G=G-1
600 T%(X,Y)=1
610 K=0
620 K=K+1
630 POKE &0177660,&0100
640 IF INP(&0177660,&0200)THEN 650
    ELSE 660
650 A%=PEEK(&0177662)
660 IF K<40*DLIT THEN 620
670 IF A%=25 THEN X=X+1
680 IF A%=8 THEN X=X-1
690 IF A%=26 THEN GOTO 790
700 IF A%=85 THEN GOTO 880
710 IF A%=27 THEN Y=Y+1
720 IF Y>21 THEN Y=0
730 IF X>21 THEN X=7
740 IF X<7 THEN X=21
750 POKE &0177660,&0000
760 GOTO 910
770 GOTO 510
780 GOTO 1020
790 PRINT AT(X,Y);"♡"
800 G=G+1
810 T%(X,Y)=1
820 Y=0
830 POKE &0177660,&0000
840 PRINT AT(X,Y);"♡"
850 G=G+1
860 T%(X,1)=0
870 GOTO 510
880 POKE &0177660,&00
890 CLEAR
900 GOTO 300
910 REM kozni
920 IF G=7*DL OR G=15*DL
    THEN 950
930 IF G=18*DL OR G=23*DL
    THEN 950
940 GOTO 510
950 FOR I%=1TO3
960 HH=8+INT(12*RND(I%*X))
970 KU=I%+INT(15*RND(HH))
980 ? AT (HH,KU);"#"
990 T%(HH,KU)=3
1000 NEXT I%
1010 GOTO 510
1020 FOR I=1 TO5
1030 CIRCLE (X*8+4,Y*10+5),
     6*I,5-I
1040 ? CHRQ(7)
1050 NEXT
1060 GOSUB 1140
1070 CLS
1080 ? AT(12,11);"bulbk !"
1090 ? "itak,wa( rezulxtat...";
     INT(G/DL)
1100 INPUT "E)e raz (D/N)";LLQ
1110 IF LLQ<>"D" THEN 1130
1120 GOTO 300
1130 END
1140 FOR I=1TO8000
1150 NEXT
1160 RETURN
```

А.Е.Кирик (г.Первомайск)

Программа POLIGON

(нашествие)

Вы совершаете полет на летающей тарелке. Необходимо разрушить все постройки, бросая бомбы. Бомбы пускают, нажимая клавишу "пробел". Ваш корабль вылетает слева и движется вправо. По достижении крайней правой позиции он вылетает слева, но уже ниже и т.д., поэтому броски должны быть точны.

Если будете тратить много зарядов, то они у вас довольно скоро закончатся. Допустим, вы разрушили все дома, тогда корабль перелетает в другой квартал. Всего 5 кварталов.

В строках 130 — 150 — вывод правил игры. 160 — 190 — если нажат символ "+", то возможно перейти к любому кварталу, иначе игра начинается с первого квартала. 110 — 280 — переход к подпрограмме, рисующей квартал. На компьютере, где была создана эта программа, началом отсчета координат является верхний левый угол экрана.

290 — 580 — основная часть программы. 690 — 910 — таблица рекордов. Дело в том, что при сбрасывании снаряда учитывается уничтоженная им площадь, таким образом вы набираете некоторое количество очков. На просьбу компьютера "Введите имя" вам следует набрать его, и тогда в таблице появится ваше имя и количество набранных вами очков, если, конечно, вы попали в число сильнейших.

Теперь о некоторых сложностях, которые могут возникнуть при написании программы на других версиях Basic.

Строка 30. В данной версии если размерность массива не превышает 10, то оператор DIM не нужен. Это следует учесть.

Строка 550. Использование операторов PRINT CHR$(27)"6": CLS: PRINT CHR$(27) "7":CLS создает эффект помех на экране монитора. Этот эффект используется при столкновении летающей тарелки с постройкой. В этой строке вы можете организовать свой эффект из операторов имеющихся в вашей версии Basic.

Операторы и функции могут выполнять другие действия: функция POINT (X, У) возвращает числовое значение в диапазоне 0..7.

0 — черный цвет, ..., 7 — белый.

Оператор LINE (X1, У1) — (X2, У2), цв, B

LINE (X1, У1) — (X2, У2), цв, BF, где X1, У1 и X2, У2 — координаты двух точек диагонали, цв — цвет рисунка, B — нарисовать прямоугольник по заданной диагонали, а добавление BF — сразу закрашивает прямоугольник. В данном компьютере число точек по X=512, по У=254. Количество символов по X=64, по У=16. Эти данные будут полезны для более точных пропорций.

```
10 REM ******** Кирик Александр  ********
20 REM ******** NASHESTVIE  v.2  ********
30 FOR V=1 TO 6 : N(V)=0 : NEXT V
40 CLS
50 PCLS
60 LOCATE,,0
70 PRINT
   "
80 PRINT
   "
                Programma


              N
A S H E S T V I E
90 PRINT
          "
                        S O N Y K

    Sostavleno 29.09.89

     Basic  Korwet  USSR
100 PRINT
          "
          programist   Kirik Alexandr

      Konegrol ul.Nakolnaj 73 .

 "
110 A$=INKEY$ : IF A$="" THEN 110
120 CLS
130 PRINT : PRINT : PRINT
          "
                  Правила игры .

           3333333333333333
"
140 PRINT
          "
Вы летите  на летающей тарелке .          Ваша задач
а          бомбы
ть все дома . Учтите что тарелка долетая до кон-
     ца квартала
выметает чуть ниже . "
150 PRINT
```

```
            "      Бо
мбм пускаются нажатием клавиши -> пробел
160 A$=INKEY$ : IF A$="" THEN 160
170 IF A$="+" THEN 180 ELSE 200
180 INPUT N$
190  N=VAL(N$)  : GOTO 210
200 N=1 : CLS
210 CLS:PCLS
220 IF N=1 THEN 230 ELSE 240
230 SS=0 : GOSUB 590
240 IF N=2 OR N=7  THEN GOSUB 610
250 IF N=3 OR N=8 THEN GOSUB 630
260 IF N=4 THEN GOSUB 650
270 IF N=5 THEN GOSUB 670
280 IF N=6 THEN GOSUB 690
290 CLS : X1=0 : X2=20 : Y1=100 : Y2=101 : LINE (1,2
50)-(512,251),1,B
: PRINT "
          Бомбы:" :
 LOCATE 14,1,0   :PRINT "Очки:"
300 LINE(X2,Y2)-(X1,Y1),,B : CIRCLE(X1+10,Y2),8,,0,3
.14
310 IF POINT(X2+2,Y2)=7 THEN 550
320 F$=INKEY$ : IF F$=" " THEN 340 ELSE 520
330 IF F$=" " THEN 340 ELSE 520
340 Y5=Y2+1 : B=B-1 : LOCATE 60,1,0
350 IF B=-1 THEN 370
360 PRINT B : GOTO 420
370 PRINT "                    У вас кончились бомбы !
" :FOR TT=1 TO 100
0 : NEXT TT
380 PRINT "                  А чего они кончились ? "
390 FOR TT=1 TO 1000 : NEXT TT
400 PRINT "                  Косить надо меньше ! ":FO
R TT=1 TO 2000 : N
EXT TT
410 GOTO 700
420 PSET(X1+10,Y5+1),7 : IF POINT(X1+10,Y6+2)=7 THEN
 430 ELSE 490
430  YY=Y6+2
440 FOR XX=1 TO 20
450 IF POINT(X1+XX,YY)=7 THEN 460 ELSE 470
460 SS=SS+1
470 NEXT XX
480 M=Y5+20 :FOR S=Y6 TO M : LINE(X1,S)-(X2,S),0 : N
EXT S :BEEP:LINE(1
,250)-(511,251),1,B : LOCATE 20,1,0 : PRINTSS : GOTO
 520
490 IF POINT(X1+10,Y6+2)=1 THEN 500 ELSE 510
500 PSET(X1+10,Y6),0 : GOTO 520
510 PSET(X1+10,Y6+1),0 : Y6=Y5+1 : GOTO 420
520 LINE(X2,Y2)-(X1,Y1),0,B : CIRCLE(X1+10,Y2),8,0,0
,3.14
530 X1=X1+4 : X2=X2+4 : IF X1>512  THEN 540 ELSE 300
540 Y1=Y1+20 : Y2=Y2+20 : X1=0 : X2=20 : IF Y2>=245
THEN 580 ELSE 300
550 FOR V=1 TO 7:PRINT CHR$(27)"6":CLS:PRINTCHR$(27)
"7":CLS:FOR TT=1 T
O 100:NEXT TT:NEXT
560 PRINT : PRINT : PRINT "                                  э
х !   Можно врезал
ись !!! "
570 FOR TT=1 TO 1000 STEP .5: NEXT TT : GOTO 700
580 N=N+1 : GOTO 210
589 REM => ********** Рисуем кварталы ************
590 LINE(50,135)-(90,249),,BF:LINE(120,155)-(160,249
),,BF:LINE(190,175
)-(230,249),,BF:LINE(260,135)-(300,249),,BF:LINE(330
,175) -(370,249),,
BF:LINE(400,155)-(440,249),,BF : B=85
600 RETURN
610 LINE(50,215)-(90,249),,BF:LINE(120,195)-(160,249
),,BF:LINE(190,175
)-(230,249),,BF:LINE(260,155)-(300,249),,BF:LINE(330
,135)-(370,249),,B
F:LINE(400,135)-(440,249),,BF : B=70
620 RETURN
630 LINE(50,135)-(90,249),,BF:LINE(120,135)-(160,249
),,BF:LINE(190,135
)-(230,249),,BF:LINE(260,135)-(300,249),,BF:LINE(330
,135)-(370,249),,B
F:LINE(400,135)-(440,249),,BF : B=100
640 RETURN
650 LINE (50,135)-(90,249),,BF : LINE(120,155)-(160,
249),,BF:LINE(190,
175)-(230,249),,BF : LINE(260,195)-(300,249),,BF : L
INE (330,215)-(370
,249),,BF : LINE (400,235)-(440,249),,BF : B=50
660 RETURN
670 LINE(50,135)-(440,249),,BF : B=120
680 RETURN
690 REM => ******** Формируем рейтинг-лист *********
700 CLS : PCLS : LOCATE 20,7 : INPUT " Введите имя";
F$
710 CLS : PCLS : LINE(5,1)-(300,250),,B :LINE(17,1)-
(17,250): LINE(5,1
)-(300,16),,B : LINE(200,250)-(300,16),,B : PRINT "
n            Имя-
             Очки "
720 G=1
730 FOR V=1 TO 6
740 G=G+2
750 LOCATE 1,G
760 PRINT V
770 NEXT V
780 O=SS : Y10=1
790 FOR V=1 TO 6
800 IF N(V)<O THEN 810 ELSE 830
810 SWAP N(V),O
820 SWAP F$(V),F$
830 NEXT V
840 FOR V=1 TO 6
850 Y10=Y10+2
860 LOCATE 7,Y10
870 PRINT F$(V)
880 LOCATE 30,Y10
890 PRINT N(V)
900 NEXT V
910 SS=0 : GOTO 160
```

А.Е.Кирик (г.Первомайск)

Программа REVERSI

Хотелось бы заметить, что компьютер делает ход не случайно, а по определенной стратегии. Программа работает достаточно быстро, несмотря на медлительность Basic.

Время обдумывания хода на компьютере CORVET порядка 14 сек. Причем каждый последующий быстрее предыдущего.

Сведения, необходимые для отладки на других компьютерах данной программы:

Строка 30. Запись: POKE &.HF72D,0 переключает клавиатуру на нижний регистр. Последующие операторы POKE формируют курсор, и их можно опустить.

Точкой отсчета координат является левый верхний угол экрана. Количество точек по горизонтали 512, по вертикали 256. Количество символов по вертикали 16, по горизонтали 64. Эти данные будут необходимы для получения более точных пропорций.

Особого внимания требуют строки 310 — 320. В строке 130 рисуется игровое поле, а в 320 идет заполнение массивов X и Y. В них содержатся координаты центров каждой клетки игрового поля.

В строках 370, 520, 570 встречается оператор PRINT CHR$(29). Код 29 — это (дубликат) аналог клавиши DEL. Использование этого оператора в цикле создает эффект сползания строки влево, т.е. удаляя надпись или сообщение, которое необходимо убрать. Эту процедуру возможно заменить командой PRINT с соответствующим числом пробелов.

Строка 2000. По адресу &HF718 записывается длительность звучания, а по адресу HF716 высота. Оператор PRINT CHR$(7) аналогичен BEEP. Эта строка вообще может содержать только RETURN.

Строки 2010 — 2090 содержат два музыкальных фрагмента. Первый фрагмент исполняется перед самым началом игры. (Строка 360; обращение: GOUB 2010), второй — при выводе сообщения о неправильном ходе (строка 520; обращение: GOSUB 2070).

Строки 2160 — 2230 — исполняется марш в честь победителя. Эти фрагменты могут быть созданы из операторов, имеющихся на вашем компьютере.

Строки 2100 — 2150 — подпрограмма вывода знака хода, о котором упоминалось в правилах игры.

```
10 REM ***************************
20 REM *   Николаевская обл.     *
30 REM *   г.Первомайск 1990г    *
40 REM *   Кирик Александр.      *
50 REM ***************************
110 DEFINT K,P,F,E,C,Z,G,A,S,T,W,H,I,X,Y,R,O,L,D,B
120 PCLS:CLS : PRINT "                                  Правила
 игры":PRINT
130 PRINT"  Игра имеет название - Reversi. Она спосо
ствует развитию
      логического мышления. На игровом поле аналогич
ному шахмат-
ной доске расположены четыре фигуры: две белые и две
 чер-"
140 PRINT"  ные. Справа находится рамка, в которую в
водятся данные о
      положении на игровом поле (соответственно коли
чество чер-
ных и белых фигур, а также количество не занятых кле
ток)."
150 PRINT"  Перед началом игры вам предстоит выбрать
 какими фигурами
      вы будете играть (белыми или черными), введя в
 компьтер
соответствующую цифру. По ходу игры на игровое поле
ста-"
160 PRINT"  вится очередная фигура. Ваша цель - набр
ать как можно боль
-     ше фигур своего цвета. Следует ставить фигуру
так, чтобы
между ней и другой вашей фигурой, уже находящейся на
 игро-
     Для продолжения нажмите <ВК>"
170 IF INKEY$="" THEN 170
180 PRINT"  вом поле, были расположены фигуры вашего
 соперника, тогда
      они становятся вашими и заменяются на фигуры в
ашего цвета
(переворачиваются). Переворачивать можно по всем нап
равле-"
190 PRINT"  ниям: по вертикали, горизонтали и по диа
гоналям. Возможны
      случаи, когда фигуры будут переворачиваться ср
азу в несколь-
ких направлениях в зависимости от расположения фигур
 супер-"
200 PRINT"  ника. Есть некоторые ограничения: делая
ход, вы должны
      обязательно учитывать, что если после вашего х
ода ни одна

210 PRINT"  вого поля ходить нельзя) нельзя пропуска
ть ход, если у вас
      есть возможность его сделать. Итак, вам предст
оит сыграть
с компьютером. Право выбирать, кто из вас (вы или ко
мпью-
      Для продолжения нажмите <ВК>"
220 IF INKEY$="" THEN 220
230 PRINT"   тер) сделает первый ход также представ
ляется вам. Стрел-
      ка указывает, кто должен делать ход. Ход задан
сывается
 двумя символами: буквой и цифрой. Например: f4. Бук
ва -"
240 PRINT"   это координата по вертикали, а цифра -
по горизонтали.
      После вас компьютер сделает обратный ход. Есл
и у вас нет
 возможности сделать ход, нажимайте клавиш <0>. Може
т сло-"
250 PRINT"   житься ситуация, когда ни у компьютера,
 ни у вас нет хода
      в этом случае нажимайте клавиш <+>. По оконча
нию игры вы-
 дается сообщение о том, кто одержал победу. Если по
желае-"
260 PRINT"   те сыграть еще раз нажмите любой клавиш
. Желаю успеха!":P
RINT:PRINT
270 PRINT "            Играете белыми или черными (1/2
) "
280 CJ$=INKEY$:IF CJ$="" THEN 280
290 CJ=VAL(CJ$):IF CJ<>1 AND CJ<>2 THEN 280
300 IF CJ=1 THEN PC=1 ELSE PC=0
310 PRINT "            Игру начинаете вы или компьютер
(1/2) "
320 CJ$=INKEY$:IF CJ$="" THEN 320 ELSE CJ=VAL(CJ$)
330 IF CJ<>1 AND CJ<>2 THEN 320
340 IF CJ=2 THEN H1=6
350 CLS:PCLS:PRINT "    Active  R      Reversi v.3
   Basic  MSX   Co
rvet ПК8010":PRINT"
    Pervomaisk,  s
chool N9.":CIRCLE(99,7),8
360 DATA 1,2,2,2,2,1,7,1,7,2,8,2,8,7,7,7,7,8,2,8,2,7
,1,7
370 FOR Y=40 TO 152 STEP 16:FOR X=35 TO 203 STEP 24:
LINE(X-12,Y-8)-(X+
12,Y+8),,B:NEXT X:NEXT Y
380  DIM H$(8),J$(8),Z(24),X(8),Y(8),CB(2),C(9,9):H$
(1)="a":H$(2)="b":
H$(3)="c":H$(4)="d":H$(5)="e":H$(6)="f":H$(7)="g":H$
(8)="h":I=0:I1=0:F

OR X=35 TO 203 STEP 24:I=I+1:X(I)=X:NEXT X:FOR Y=40
TO 152 STEP 16:I1=
I1+1:Y(I1)=Y:NEXT Y:CIRCLE(X(4),Y(4)),6,6,,,1.3
390 J$(1)="1":J$(2)="2":J$(3)="3":J$(4)="4":J$(5)="5
":J$(6)="6":J$(7)=
"7":J$(8)="8"
400 CIRCLE(X(5),Y(5)),6,6,,,1.3:CIRCLE(X(5),Y(4)),6,
6,,,1.3:PAINT(X(5)
,Y(4)),7,6:CIRCLE(X(4),Y(5)),6,6,,,1.3:PAINT(X(4),Y(
5)),7,6:DIM A(9,9)
,B(8,8): A(4,4)=1 : A(5,4)=7 : A(5,5)=1 : A(4,5)=7
410 FOR G=1 TO 24:READ Z(G):NEXT G
420 PU=60:CH=2:BE=2:LOCATE,,1:FOR J=8 TO 1 STEP -1:P
RINT "";J:NEXT:PRI
NT"    a  b  c  d  e  f  g  h":PRINT:PRINT " Вам ход
 :":PRINT " Ход ЭВ
M :";:GOSUB 2470:GOSUB 2070:IF H1=6 THEN 630
430 GOSUB 2160:LOCATE 12,13,1:FOR T=1 TO 15:PRINT CH
R$(29);:NEXT
440 KX$=INKEY$ : IF KX$="" THEN 440
450 IF KX$<>"0" THEN 470
460 GOSUB 2380:IF H1=1 THEN 440 ELSE PRINT"Пропуск х
ода":GOTO 630
470 IF KX$<>"+" THEN 510
480 PD=PC:PC=0:GOSUB 2380:IF H1=0 THEN 500
490 PC=PD:GOTO 440
500 PC=1:GOSUB 2380:IF H1=1 THEN 440: PX=PU:PU=0:TR=
1:GOSUB 2300
```

```
510 KX=ASC(KX$)-96:IF KX<1 OR KX>8 THEN 430
520 LOCATE 12,13: PRINT KX$
530 LOCATE 13,13:KY$=INKEY$:IF KY$="" THEN530 ELSE K
Y=9-VAL(KY$)
540 IF KY<1 OR KY>8 THEN 530
550 LOCATE 13,13,0: PRINT KY$
560 REM => %%%%%% Проверка на правильный ход %%%%%%
%%%%%%%%%%
570 IF A(KX,KY)=0 THEN 590
580 LOCATE 12,13: PRINT " Сюда нельзя !":GOSUB 2130:
FOR T=27 TO 12 STE
P -1:LOCATE 12,13:PRINTCHR$(29);:FOR S=1 TO 20:NEXT
S:NEXT T:GOTO 430
590 IF PC=0 THEN 610 : REM  => %%%%% Ход человека %%
%%%%%%%%%
600 CB=1:CB(1)=1:CB(2)=7:Z=7:ZI=7:GOTO 620
610 CB=7:CB(1)=7:CB(2)=1:Z=0:ZI=1
620 ERASE C:DIM C(9,9):E=0:POLZ=1:H=1:GOSUB900:GOSUB
2300:IF E=1 THEN58
0
630 LOCATE12,14:GOSUB 2200:FOR T=1 TO 15:PRINT CHR$(
29);:NEXT T:IF PC
=0 THEN 640 ELSE 650 : REM => Ход компьютера
640 CB=1:CB(1)=1:CB(2)=7:Z=7:ZI=7:GOTO 660
650 CB=7:CB(1)=7:CB(2)=1:Z=0:ZI=1
660 LOCATE 14,12:PRINT"                          ":POLZ=2:
ERASE C:DIM C(9,9)
:H=0:HO=0
670 FOR KY=1 TO 8:FOR KX=1 TO 8:IF A(KX,KY)=0 THEN 6
80 ELSE 690
680 GOSUB 900
690 NEXT KX:NEXT KY
700 IF C(1,1)>HO THEN HO=C(1,1) ELSE 720
710 KX=1 : KY=1:FOR G=1 TO 6 :  Z(G)=0 : NEXT G
720 IF C(8,8)>HO THEN HO=C(8,8) ELSE 740
730 KX=8 : KY=8:FOR G=7 TO 12: Z(G)=0 :NEXT G
740 IF C(8,1)>HO THEN HO=C(8,1) ELSE 760
750 KX=8 : KY=1:FOR G=13 TO 18: Z(G)=0 : NEXT G
760 IF C(1,8)>HO THEN HO=C(1,8) ELSE 780
770 KX=1 : KY=8:FOR G=19 TO 24: Z(G)=0 : NEXT G
780 IF HO>0 THEN 880 ELSE GOSUB 2520
790 IF HO>0 THEN 880
800 FOR PO=1 TO 23 STEP 2:E(Z(PO),Z(PO+1))=C(Z(PO),Z
(PO+1)):C(Z(PO),Z(
PO+1))=0:NEXT PO
810 FOR F=1 TO 8 : FOR W=1 TO 8 : IF C(W,F)>HO THEN
HO=C(W,F) ELSE 830
820 KX=W : KY=F
830 NEXT W : NEXT F : IF HO=0 THEN 840 ELSE 880
840 FOR PO=1 TO 23 STEP 2:IF E(Z(PO),Z(PO+1))<HO THE
N 860
850 HO=E(Z(PO),Z(PO+1)):KX=Z(PO) : KY=Z(PO+1)
860 NEXT PO : IF HO>0 THEN 880
870 LOCATE 12,14:PRINT"                          ":LOCATE
12,14 : PRINT " Я
пропускаю ход" : GOTO 430
880 REM => %%%%%%%%%%%%%%% Вывод хода на ЗКУ %%%%%%
%%%%%%%%%%
890 LOCATE 12,14:PRINT"                          ":LOCAT
E 12,14 : PRINT H$
(KX)+J$(9-KY);:H=1:POLZ=1:GOSUB 900:GOSUB 2300:GOTO
430
900 REM => %%%%%%%%%%%%%%%% Подпрограмма БЛОУК %%%%%%
%%%%%%%%%%
910 IF KX-1>0 AND A(KX-1,KY)=CB THEN GOSUB 1010
920 IF KY-1>0 AND A(KX,KY-1)=CB THEN GOSUB 1130
930 IF KX+1<9 AND A(KX+1,KY)=CB THEN GOSUB 1250
940 IF KY+1<9 AND A(KX,KY+1)=CB THEN GOSUB 1370
950 IF KX-1>0 AND KY-1>0 AND A(KX-1,KY-1)=CB THEN GO
SUB 1490
960 IF KX+1<9 AND KY-1>0 AND A(KX+1,KY-1)=CB THEN GO
SUB 1610
970 IF KX+1<9 AND KY+1<9 AND A(KX+1,KY+1)=CB THEN GO
SUB 1730
980 IF KX-1>0 AND KY+1<9 AND A(KX-1,KY+1)=CB THEN GO
SUB 1850
990 IF C(KX,KY)=0 AND POLZ=1 THEN E=1
1000 RETURN
1010 P=0 : REM => Подпрограмма - 1
1020 FOR I=1 TO 6
1030 IF KX-I>0 THEN 1050 ELSE P=0
1040 I=1: C(KX,KY)=C(KX,KY)+P : IF H=1 THEN 1100 ELS
E  RETURN
1050 IF A(KX-I,KY)=CB(1) THEN P=P+1 ELSE 1070
1060 NEXT I
1070 IF KX-I>0 THEN 1080 ELSE 1090
1080 IF A(KX-I,KY)=CB(2) THEN 1040
1090 P=0 : GOTO 1040
1100 IF P=0 THEN RETURN
1110 OB=1:PX=KX : PY=KY : GOSUB 1970
1120 OB=2:FOR I=1 TO P : PX=PX-1 : GOSUB 1970 : NEXT
 I : RETURN
1130 P=0 : REM => Подпрограмма - 2
1140 FOR I=1 TO 6
1150 IF KY-I>0 THEN 1170 ELSE P=0
1160 I=1: C(KX,KY)=C(KX,KY)+P :IF  H=1 THEN 1220 ELS
E  RETURN
1170 IF A(KX,KY-I)=CB(1) THEN P=P+1 ELSE 1190
1180 NEXT I
1190 IF KY-I>0 THEN 1200 THEN 1210
1200 IF A(KX,KY-I)=CB(2) THEN 1160
1210 P=0 : GOTO 1160
1220 IF P=0 THEN RETURN
1230 OB=1:PX=KX : PY=KY : GOSUB 1970
1240 OB=2:FOR I=1 TO P : PY=PY-1 : GOSUB 1970 : NEXT
 I : RETURN
1250 P=0 : REM => Подпрограмма - 3
1260 FOR I=1 TO 6
1270 IF KX+I<9 THEN 1290 ELSE P=0
1280 I=1: C(KX,KY)=C(KX,KY)+P : IF H=1 THEN 1340 ELS
E  RETURN
1290 IF A(KX+I,KY)=CB(1) THEN P=P+1 ELSE 1310
1300 NEXT I
1310 IF KX+I<9 THEN 1320 ELSE 1330
1320 IF A(KX+I,KY)=CB(2) THEN 1280
1330 P=0 : GOTO 1280
1340 IF P=0 THEN RETURN
1350 OB=1:PX=KX : PY=KY : GOSUB 1970
1360 OB=2:FOR I=1 TO P : PX=PX+1 : GOSUB 1970 : NEXT
 I : RETURN
1370 P=0 : REM => Подпрограмма - 4
1380 FOR I=1 TO 6
1390 IF KY+I<9 THEN 1410 ELSE P=0
1400 I=1: C(KX,KY)=C(KX,KY)+P : IF H=1 THEN 1460 ELS
E  RETURN
1410 IF A(KX,KY+I)=CB(1) THEN P=P+1 ELSE 1430
1420 NEXT I
1430 IF KY+I<9 THEN 1440 ELSE 1450
1440 IF A(KX,KY+I)=CB(2) THEN 1400
1450 P=0 : GOTO 1400
1460 IF P=0 THEN RETURN
1470 OB=1:PX=KX : PY=KY : GOSUB 1970
1480 OB=2:FOR I=1 TO P : PY=PY+1 : GOSUB 1970 : NEXT
 I : RETURN
1490 P=0 : REM => Подпрограмма - 5
1500 FOR I=1 TO 6
1510 IF KX-I>0 AND KY-I>0 THEN 1530 ELSE P=0
1520 I=1: C(KX,KY)=C(KX,KY)+P : IF H=1 THEN 1580 ELS
E  RETURN
1530 IF A(KX-I,KY-I)=CB(1) THEN P=P+1 ELSE 1550
1540 NEXT I
1550 IF KX-I>0 AND KY-I>0 THEN 1560 ELSE 1570
1560 IF A(KX-I,KY-I)=CB(2) THEN 1520
1570 P=0 : GOTO 1520
1580 IF P=0 THEN RETURN
1590 OB=1:PX=KX : PY=KY : GOSUB 1970
1600 OB=2:FOR I=1 TO P : PX=PX-1 :PY=PY-1:GOSUB 1970
 : NEXT I : RETURN
1610 P=0 : REM => Подпрограмма - 6
1620 FOR I=1 TO 6
1630 IF KX+I<9 AND KY-I>0 THEN 1650 ELSE P=0
1640 I=1: C(KX,KY)=C(KX,KY)+P:IF H=1 THEN 1700 ELSE
RETURN
1650 IF A(KX+I,KY-I)=CB(1) THEN P=P+1 ELSE 1670
1660 NEXT I
1670 IF KX+I<9 AND KY-I>0 THEN 1680 ELSE 1690
1680 IF A(KX+I,KY-I)=CB(2) THEN 1640
1690 P=0 : GOTO 1640
1700 IF P=0 THEN RETURN
1710 OB=1:PX=KX : PY=KY : GOSUB 1970
1720 OB=2:FOR I=1 TO P : PX=PX+1 : PY=PY-1 : GOSUB 1
970 : NEXT I: RETU
RN
1730 P=0 : REM => Подпрограмма - 7
1740 FOR I=1 TO 6
1750 IF KX+I<9 AND KY+I<9 THEN 1770 ELSE P=0
1760 I=1: C(KX,KY)=C(KX,KY)+P:IF H=1 THEN 1820 ELSE
RETURN
1770 IF A(KX+I,KY+I)=CB(1) THEN P=P+1 ELSE 1790
1780 NEXT I
1790 IF KX+I<9 AND KY+I<9 THEN 1800 ELSE 1810
1800 IF A(KX+I,KY+I)=CB(2) THEN 1760
1810 P=0 : GOTO 1760
1820 IF P=0 THEN RETURN
1830 OB=1:PX=KX : PY=KY : GOSUB 1970
1840 OB=2:FOR I=1 TO P : PX=PX+1 : PY=PY+1 : GOSUB 1
970 : NEXT I : RET
URN
1850 P=0 : REM => Подпрограмма - 8
1860 FOR I=1 TO 6
1870 IF KX-I>0 AND KY+I<9 THEN 1890 ELSE P=0
1880 I=1: C(KX,KY)=C(KX,KY)+P : IF H=1 THEN 1940 ELS
E  RETURN
1890 IF A(KX-I,KY+I)=CB(1) THEN P=P+1 ELSE 1910
1900 NEXT I
1910 IF KX-I>0  AND KY+I<9 THEN 1920 ELSE 1930
1920 IF A(KX-I,KY+I)=CB(2) THEN 1880
1930 P=0 : GOTO 1880
1940 IF P=0 THEN RETURN
1950 OB=1:PX=KX : PY=KY : GOSUB 1970
1960 OB=2:FOR I=1 TO P : PX=PX-1 : PY=PY+1 : GOSUB 1
970 : NEXT I : RET
URN
```

```
1970 REM => %%%%%%%%%%%%%%%%%% Рисование фигур %%%%
%%%%%%%%%%
1980 IF OB=2 THEN 2020
1990 IF A(KX,KY)>0 AND OB=1 THEN RETURN
2000 IF ZI=1 THEN CH=CH+1  ELSE BE=BE+1
2010 GOTO 2030
2020 IF ZI=1 THEN CH=CH+1:BE=BE-1 ELSE BE=BE+1:CH=CH
-1
2030 FOR RA=5 TO 0 STEP -1
2040 A(PX,PY)=ZI : CIRCLE(X(PX),Y(PY)),RA,Z,,,1.3
2050 NEXT RA : CIRCLE(X(PX),Y(PY)),6,6,,,1.3
2060 POKE &HF718,2:POKE &HF716,35:PRINT CHR$(7);:RET
URN
2070 POKE &HF718,7:REM => Музыкальные эффекты
2080 FOR I=7 TO 25:POKE &HF716,I:PRINT CHR$(7);:NEXT
 I
2090 FOR I=20 TO 11 STEP -1:POKE &HF716,I:PRINT CHR$
(7);:NEXT I
2100 FOR T=1 TO 3:FOR G=1 TO 25:NEXT G
2110 FOR K%=7 TO 3 STEP -1:POKE &HF716,K%:PRINT CHR$
(7);:NEXT K%
2120 NEXT T : RETURN
2130 POKE &HF718,18:FOR T=1 TO 2
2140 FOR I=11 TO 19 STEP 2:POKE &HF716,I:PRINT CHR$(
7);:NEXT I
2150 NEXT T:RETURN
2160 REM => %%%%%%%%%%%% Знак хода %%%%%%%%%%%%%%%
%%%%%%%%
2170 M=0:FOR X=215 TO 220:LINE(X,200+M)-(X,200-M):LI
NE(X,216+M)-(X,216
-M),0:M=M+1:NEXT
2180 Y1=197:Y2=Y1+6:Y3=213:Y4=Y3+6
2190 FOR X=220 TO 240:LINE(X,Y1)-(X,Y2),7:LINE(X,Y3)
-(X,Y4),0:NEXT X:R
ETURN
2200 M=0:FOR X=215 TO 220:LINE(X,216+M)-(X,216-M):LI
NE(X,200+M)-(X,200
-M),0:M=M+1:NEXT X:Y1=197:Y2=Y1+6:Y3=213:Y4=Y3+6

2210 FOR X=220 TO 240:LINE(X,Y3)-(X,Y4):LINE(X,Y1)-(
X,Y2),0:NEXT X:RET
URN
2220 POKE&HF718,20: REM => Музыкальные эффекты
2230 FOR L1=1 TO 2
2240 FOR I=8 TO 10:POKE &HF716,I:PRINT CHR$(7);:NEXT
 I:FOR T=1 TO 70:N
EXT:FOR L=1 TO 50:NEXT L
2250 FOR T=1 TO 2:POKE&HF716,22:PRINT CHR$(7);:FOR L
=1 TO 40:NEXT L:NE
XT T:NEXT L1
2260 FOR D=1 TO 3:POKE&HF716,16:FOR L=1 TO 100:NEXT
L:PRINTCHR$(7);:PO
KE&HF716,12:PRINTCHR$(7);:NEXT D:FOR L=1 TO 50:NEXT
L
2270 FOR D=1 TO 2:POKE&HF716,20:PRINTCHR$(7);:FOR L=
1 TO 80:NEXT L:NEX
T D
2280 FOR I=30 TO 10 STEP- 5:POKE &HF716,I:PRINTCHR$(
7);:NEXT I
2290 RETURN
2300 REM => %%%%%%%% Подсчет фигур %%%%%%%%%%%%%%%
%%%%%%%%%%
2310 PU=64-BE-CH:LOCATE 50,6:PRINTCH:LOCATE 50,7:PRI
NTBE:LOCATE 50,8:I
F TR=1 THEN PRINTPX ELSE PRINTPU
2320 IF BE=0 OR CH>BE AND PU=0 THEN R$="черные!"
2330 IF CH=0 OR CH<BE AND PU=0 THEN R$="белые!"
2340 IF CH=BE AND PU=0 THEN R$="разум и находчивость
!"
2350 IF R$="" THEN RETURN
2360 LOCATE 40,13:PRINT"Победили "+R$:LOCATE 14,1:GO
SUB 2220
2370 IF INKEY$="" THEN 2370 ELSE RUN
2380 REM => Подпрограмма ПОХ %%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%
2390 H1=0 : IF PC=0 THEN 2410
2400 CB=1:CB(1)=1:CB(2)=7:GOTO 2420
2410 CB=7:CB(1)=7:CB(2)=1
2420 ERASE C:DIM C(9,9):H=0:POLZ=2
2430 FOR KY=1 TO 8:FOR KX=1 TO 8:IF A(KX,KY)<>0 THEN
 2460
2440 GOSUB 900:IF C(KX,KY)=0 THEN 2460 ELSE H1=1
2450 RETURN
2460 NEXT KX:NEXT KY:RETURN
2470 REM => %%%%% Ранка %%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%
2480 LOCATE 35,5:PRINT" ":LOCATE 35,6:
          PRINT"                    ":LOCATE 35,7:

     PRINT"                    ":LOCATE 35,8:

 PRINT"                    "
2490 LOCATE 35,9:                    PRINT" "
2500 LOCATE 40,6:PRINT "Черные =>";:LOCATE 40,7:PRIN
T"Белые  =>";:LOCA
TE 40,8:PRINT"Пустые =>";
2510 LOCATE 50,6:PRINTCH:LOCATE 50,7:PRINTBE:LOCATE
50,8:PRINTPU:RETUR
N
2520 REM => %%%%%%%%%%% Подпрограмма DM-1 %%%%%%%%
%%%%%%%%%%
2530 DATA 1,2,2,2,2,1
2540 DATA 7,1,7,2,8,2
2550 DATA 8,7,7,7,7,8
2560 DATA 1,7,2,7,2,8
2570 IF A(1,1)>0 THEN RESTORE 2530:GOSUB 2620
2580 IF A(8,1)>0 THEN RESTORE 2540:GOSUB 2620
2590 IF A(8,8)>0 THEN RESTORE 2550:GOSUB 2620
2600 IF A(1,8)>0 THEN RESTORE 2560:GOSUB 2620
2610 RETURN
2620 FOR T=1 TO 3:READ CX,CY:IF C(CX,CY)>HO THEN KX=
CX:KY=CY:HO=C(KX,K
Y)
2630 NEXT T:RETURN
```


ЛЕНИНГРАДСКИЙ
ПРОИЗВОДСТВЕННЫЙ КООПЕРАТИВ
"ОСТ"
ПРЕДЛАГАЕТ

программное обеспечение для компьютеров
БК-0010, 0010.01, ДВК-2М, 3М, КУВТ-86, УКНЦ
и для всех совместимых с IBM PC

♣ Кооператив располагает обширным банком игровых, учебных, системных, прикладных программ.

♣ Расценки значительно ниже государственных.

♣ Списки программ при указании типа ЭВМ высылаются бесплатно.

Адрес: 199034, Ленинград, а/я 437.

Л.В.Жданов (г.Брянск)

Программа TAS

В журнале "Наука и жизнь" N 2 за 1990 год была опубликована программа на БЕЙСИКе "Пасьянс "Турецкий платок". На БК (да и на IBM PC — тоже) эта программа работает медленно и в управлении тяжела.

Мне понравилось, что в ней не надо стучать по клавишам. Таким свойством обладают не так уж много игр: на БК это шахматы, крестики-нолики и реверси.

Я написал игру для ДВК и БК на ассемблере, добавив подсказки, сдвиг столбца вниз, отмену ходов и гадание (описано в "Комсомольской правда" в декабре 90 года).

Потом оказалось, что есть аналогичная игра (для IBM PC) — MAHJONGG. Там великолепная графика, а в остальном моя программа лучше.

Какая может быть польза от этой игры.

- Потратив некоторое время на игру, можно придумать тьму задач по теории вероятностей.
- Возможно, поможет найти критерий сходимости пасьянса. А это может дать эффект в совсем другой области.
- Тренировка внимания и работоспособности (как и все игры).

Кроме того, я считаю, что в гадании нет никакой мистики: карты помогают продумать ситуации, которые и в голову не приходили.

Мне удалось найти простой алгоритм тасовки колоды во время опроса клавиатуры и джойстика. (Не понятно, зачем во многих играх управление от джойстика задается отдельно.)

Вот текст подпрограммы TAS.

```
      RK=177660              ; Регистр состояния клавиатуры
      RD=177662              ; Регистр данных клавиатуры
      RJ=177714              ; Порт ввода/вывода (джойстик)
      .....
TAS:  BIS #100,@#RK          ; Запрет прерываний
      TSTB @#RD              ; Сброс готовности клавиатуры
      CLR R0                 ; \
      CLR R1                 ;  ]
 T1:  INC R0                 ;  ]
      CMP #64,R0             ;  ]    Тасовка. Если не прерывать, даст около
      BNE T2                 ;  ]  8000000 раскладов. После нажатия клавиши
      CLR R0                 ;  ]  будет гнать по другому кольцу такой же
 T2:  ADD R0,R1              ;  ]  длины, т.е. раскладов будет много.
 T3:  CMP R1,#56             ;  ]   Любой ли расклад достижим?
      BLE T4                 ;  ]
      SUB #57,R1             ;  ]
      BR T3                  ;  ]
 T4:  MOVB KLD(R1),R2        ;  ]   KLD - массив номеров карт 0..51(дес)
      MOVB KLD(R0),KLD(R1)   ;  ]
      MOVB R2,KLD(R0)        ; /
      CMPB @#RJ,L            ; Джойстик трогали?
      BEQ MK                 ; Если нет, то посмотрим клавиатуру
 TJ:  MOVB @#RJ,L            ; Подпрограмма служит для получения упр.байта L
      CMPB L,DL              ; Джойстик в нейтральном положении?
      BEQ MK                 ;
      MOV #3000,R2           ; \
TJJ:  CMPB @#RJ,L            ;  ] Обработка дребезга джойстика
      BNE TJ                 ;  ]
      SOB R2,TJJ             ; /
      RTS PC                 ;
 MK:  TSTB @#RK              ; Если клавиша не нажата, продолжать тасовку
      BPL T1                 ;
DR1:  MOV #3000,R0           ; \
      TSTB @#RD              ;  ]
DR2:  TSTB @#RK              ;  ] Обработка дребезга клавиатуры
      BMI DR1                ;  ]
      SOB R0,DR2             ; /
      MOVB @#RD,L            ;
      BICB @#240,L           ; Верхний латинский регистр
      RTS PC                 ;
```

ГИПЕРОН

Внешнеэкономическая фирма "Гиперон" приглашает к сотрудничеству!

"Гиперон" представляет целостную систему с широко разветвленной сетью брокерских представительств на Московской центральной фондовой бирже, Московской товарной и Российской товарно-сырьевой биржах, Международной бирже электроники, связи, телекоммуникаций, радио- и функционального приборостроения.

"Гиперон" поможет Вам быстро продать и с выгодой купить любые товары, в любое время представит информацию о конъюнктуре рынка, наличии конкретного товара, текущих биржевых ценах, окажет услуги при осуществлении бартерных операций.

"Гиперон" предоставит комплекс услуг по вопросам перевода предприятий и организаций на новые формы хозяйствования и собственности, поможет Вам успешно осуществить переход на акционерную форму собственности. В рамках этой работы фирма:

♣ *разработает концепцию перехода на акционерную форму собственности;*
♣ *подготовит учредительные документы для государственной регистрации;*
♣ *проведет экспертную оценку хозяйственно-финансового положения предприятия;*
♣ *проведет в составе комиссии оценку имущества государственного предприятия;*
♣ *обеспечит бланками ценных бумаг, изготовленными лучшими зарубежными фирмами с многократной системой защиты;*
♣ *осуществит первичное размещение ценных бумаг эмитента (акций, облигаций, депозитных сертификатов), проведет операции по их купле-продаже.*

ВНЕШНЕЭКОНОМИЧЕСКАЯ ФИРМА "ГИПЕРОН" — ВАШ НАДЕЖНЫЙ ПАРТНЕР!

Заказы и предложения по сотрудничеству направляйте по адресу:
123436, Москва, 1-й Волоколамский проезд, д.10.
Телефон: 196-68-29, 194-05-33
Телефакс: 196-82-38

Дм.Новиков (г.Троицк)

"Питон"

Программа реализуется на микрокомпьютере МК-85 (где есть псевдографика, операторы DRAW и DRAWC). Это аналог игры "Питон" или "Удав", знакомой пользователям ПЭВМ на 580-м процессоре.

О работе программы. В начале игры сам "Питон" изображается на экране бегущей точкой. Управление точкой осуществляется нажатием клавиш 1:9. Задача игрока, совмещая управляемую точку с появляющимися на экране другими точками — "лягушками" (которые еще и "прыгают" с места на место), затереть их все. С каждой затертой точкой "питон" увеличивается на одну клеточку, но по мере продвижения по экрану уменьшается (при этом появляются новые "лягушки"), и если вовремя ему не "подкрепиться", "питон" может совсем сойти на нет — "умереть с голоду".

В данной игре "питону" дозволяется переползать через самого себя и менять направление на обратное. За предел экрана он не выходит, как бы "отражаясь" от края индикатора.

В игре может быть несколько уровней (в приводимом случае три, контролируются переменной Q), каждый следующий предусматривает быстрейшее "похудание" "питона" и как следствие этого — требование более экономичной охоты за "лягушками".

Игрок, прошедший все уровни, побеждает.

Программа занимает около 880 байт (шагов) в памяти и использует все 26 ячеек памяти (от A до Z).

От нажатия случайных клавиш программа защищена.

Запускать программу следует в режиме повышенного быстродействия.

Основная программа
(См."Блок подпрограмм")

```
1  VAC:Q=Q+1
2  IF Q ≥4; PRINT "Вы выиграли" : END
3  PRINT "Уровень"; Q; : FOR Z TO 99:
NEXT Z : CSR0,
4  R=1:X=2:FOR Y=4 TO 14 STEP2
5  A(Y)=INT (RAN#*6):B(Y) = INT
(RAN#*59)
6  DRAW B(Y), A(Y): IF R=1; NEXT Y
7  FOR R=0 TO 99/X STEP Q : FOR Y =0
TO X STEP 2:FOR U=0 TO X STEP 2
8 IF Y ≠ U:IFA(U)=A(Y); IF B(U)=B(Y)
THEN 11
9 NEXT U:FOR U=X+2 TO 14 STEP
2:IFA(U)=A(Y):IF B(U)=B(Y):THEN 11
10 NEXT U: DRAWC B(Y); A(Y)
11 $ KEY: IF$≠""; GOSUB 33
12 FOR Z= 0 TO 1 : Y=Y+Z:IF Y=Z;
A(Y)=A(Y+X-T)+V(Z):IFZ=1; T=0
13 IF Y≠Z; A(Y)=A(Y-2)+V(Z)
14 IF A(Y)>6+Z*53; V(Z)= -V(Z):A(Y) =
A(Y)-1
15 IF A(Y) < 0 ; V(Z)=-V(Z):A(Y) = A(Y)+1
16 NEXT Z:Y=Y-1
17 IF T=2 THEN 21
18 S = X+2 : FOR U=STO 14 STEP 2
19 IF A(Y)=A(U); IF B(Y)=B(U); GOSUB 27
20 NEXT U:IF X ≥ 14 THEN 2
21 DRAW B(Y), A(Y)
22 NEXT Y
23 U=INT (RAN#*8)*2:IF U ≥S+T; GOSUB
28
24 NEXT R: Y=X: X=X-2: IF X ≥2 THEN 5
25 PRINT "Умер";
26 END
```

Блок подпрограмм

```
27 T=2: X=S: B(U)=B(X):A(U) =
A(X):B(X) = -1:RETURN
28 DRAW B(U), A(U)
29 FOR Z=0 TO 1:U=U+Z:A(U) =
A(U)+SGN(RAN#-0.5)
30 IF A(U)< 0 : A(U)=1
31 IF A(U) 6 + 53*Z; A(U)=A(U)-2
32 NEXTZ: U=U-1 : DRAW B(U),
A(U):RETURN

33 IF ASCI S 57; RETURN
34 IF ASCI S 49; RETURN
35 GOSUB ASCI $ : RETURN

49 W=-1:V=-1: RETURN
50 W=0 : V=-1 : RETURN
51 W=1 : V=-1 : RETURN
52 W=-1: V= 0 : RETURN
53 W= 0: V= 0 : RETURN
54 W=1 : V= 0 : RETURN
55 W=-1: V= 1 : RETURN
56 W= 0: V= 1 : RETURN
57 W= 1: V= 1 : RETURN
```

В.Зорин (г.Рига)

Угадай число

В № 2 1991 г. была напечатана программа для ПМК — "Угадай число".

```
10 PRINT "УГАДАЙ ЧИСЛО"
20 T = 0: N = INT(RAN#*100)
30 T = T + 1
40 IF T = 8 THEN 120
50 PRINT "П"; T;
60 CSR 7: INPUT G
70 IF G = N THEN 110
80 IF G > N THEN 100
90 PRINT G; "Мало": GOTO 30
100 PRINT G; "Много": GOTO 30
110 PRINT "Ты угадал": GOTO 140
120 PRINT "Все попытки исчерпаны,";
130 PRINT "а число было"; N
140 PRINT "Сыграем еще?": GOTO 20
```

Предлагаю еще один (как считаю, улучшенный) вариант этой игры. Программа занимает 74 шага и 11 регистров памяти. Для разработки использовался программируемый микрокалькулятор МК-52.

	00	10	20	30	40	50	60	70
0	0Г	40	69	Г4	5С	11	01	01
1	44	49	40	64	20	01	51	11
2	63	64	68	65	51	10	65	0L
3	50	0Е	23	50	02	62	53	15
4	40	0С	35	4Е	66	51	69	52
5	59	0L	48	5С	11	39	5Г	
6	02	01	67	44	5Е	5С	33	
7	5Е	5Г	12	66	56	63	51	
8	11	13	34	61	69	53	37	
9	05	45	46	50	60	69	60	

После ввода программы в ПМК необходимо занести исходные данные:

в R1 8808888 В ↑ 2204632 К V К <x > ВП 5
в R2 888888 В ↑ 665626 К V R <x> ВП 5
в R3 888808 В ↑ 66770299 К V К <x> ВП 1
в R7 100
в R8 число в промежутке от 0 до 1, не включая их самих.

Программа начинает работать, и на экране появляется заставка:

Е. 0 — 99

что означает: вам необходимо набрать число попыток, и нажать С/П. Если вы ввели 0, то ПМК сам вам дает 5 попыток.

Чем больше число, тем дольше ПМК будет его обрабатывать.

ПРИМЕР:

1 попытка		8 секунд	
3	— — # — —	12	— — # — —
5	— — # — —	15	— — # — —
10	— — # — —	22	— — # — —
20	— — # — —	38	— — # — —

После этого на экране появится информация в виде:

Е. — не значащий символ;

0 — 99 — диапазон загадываемого числа;

-xx — кол-во попыток;

а в регистре V (↔) номер игры.

Далее начинается игра. Вы вводите число и нажимаете С/П. Примерно через 7 секунд ПМК выдаст информацию в виде:

-1. — xx,

что означает: ваше число мало и у вас осталось <xx> попыток. Или

1. -xx,

что означает: ваше число велико и у вас осталось <xx> попыток. Если вы угадали число, то на экране —

ЕГЕ-Е.,

а в регистре Y (↔), за сколько вы угадали это число. Ну а если ваши попытки закончились, то на экране появится такая заставка:

- OCEL. —

и в регистре Y (↔) задуманное ПМК число.

Если на заставку ЕГЕ-Е или -OCEL. — нажать С/П, то ПМК загадает новое число и через 6 секунд опять готов к работе. Но если на эти заставки нажать /-/, а потом С/П, то на экране появится надпись

Е. 0-99

и вы должны будете ввести количество попыток (см. начало игры). Если во время игры ввести отрицательное число, то это означает поражение, и на экран выдается -OCEL. — , и в регистре Y (↔) — задуманное число.

ОПИСАНИЕ РЕГИСТРОВ

R0 — ОПЕРАТИВНЫЙ, служит для кол-ва повторений.

R1 — заставка (OCEL.-)

R2 — заставка (ЕГЕ-Е.)

R3 — заставка (Е. 0-99)

R4 — счетчик игр.

R5 — хранение заставки с кол-вом ходом (Е. 0-99 -xx)

R6 — задуманное калькулятором число.

R7 — ограничитель задуманного числа (100).

R8 — число в промежутке от 0 до 1. Используется в качестве генератора случайных чисел.

R9 — содержит кол-во ходов.

Re — ОПЕРАТИВНЫЙ, временное хранение числа игрока.

Свободные регистры памяти: Ra, Rb, Rc, Rd.

В.Воронихин, А.Туманов
(г.Москва)

"Поймай точку"

На дисплее справа и слева падают точки. Их нужно ловить, подставляя под них ракетку, которая управляется клавишами Z — влево, EE — вправо, СТОП — остальные. Перед началом игры калькулятор запрашивает ваш ранг. Чем меньше ранг, тем более высок темп игры. Советуем начинать с ранга 4 или 5. Каждая непойманная точка отображается в левом верхнем углу. Каждые десять пойманных — точкой в правом углу. При числе пойманных точек, кратном 100, число непойманных уменьшается на 5 или полностью сбрасывается, если оно меньше 5.

```
1 A=0: B=0: C=0: PRINT "(c) Malysh"
10 INPUT "Range", P: PRINT "_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _"
20 H=0: F=0: M=30
30 K=RAN# * 34 + 13 : IF H=10 THEN 180
40 DRAW K,6: FOR N=0 TO 5
50 IF I=90; M=M-5:T=5
60 IF I=123; M=M+5: T=-5
70 FOR L=0 TO 4 : DRAWC M+T+L, 0:DRAW M+L, 0
80 FOR Z=0 TO _ P: S=ASCI KEY: IF S≠0; I=5
90 NEXT Z: NEXT L: DRAW K, 5-N: DRAWC K, 6-N: NEXT N
100 IF ≥M; IF K < M+5 THEN 120
110 DRAW H, 6:H=H+1 : GOTO 30
120 F=F+1 : IF F/100≠INT (F/100) THEN 160
130 IF P > 0; P=P-1
140 H=H-5 : IF H<0; H=0
150 FOR S=H TO H+4 : DRAWC S,6 : NEXT S : DRAW 50,F/100-1
160 IF F/10=INT(F/10); DRAW 60-(F/10-10*INT(F/100)),F/100
170 GOTO 30
180 PRINT "Your Result= "; F
190 IF F > C; C=F
200 IF F > B; C=B : B=F
210 IF F > A: B=A : A=F
220 PRINT A; "  "; B; "  "; C: GOTO 10
```

Не забудьте, включая калькулятор, держать нажатой клавишу "+".

Заключение рецензента

Игра использует возможности индикатора МК-85 по созданию движущихся точек и тире. Задача игрока, используя тире, преградить дорогу точкам, движущимся по вертикали. Этим она несколько напоминает известную детскую игру "Ну погоди!", вмонтированную в будильник с ЖКИ. Не сомневаюсь в том, что эта игра вызовет не только другие подражательные игры, но и новые игры, использующие не полностью раскрытые сейчас возможности ЖКИ.

В.А.Хмелюк

"Слалом"

Игра "Слалом" (В.В.Ермаков) впервые напечатана в № 8 за 1988 г.

С некоторым опозданием поступила рецензия на эту игру. Откликов от читателей на опубликованную программу было много, поэтому приводим полностью текст рецензии и листинг отредактированной программы.

```
5 REM *** СЛАЛОМ ***
6 REM БЕЙСИК БК0010.01
7 Автор - В.В.Ермаков
10 CLS
20 A$=CHR$(182%)
30 B$=CHR$(127%)
50 ? AT(3%,4%)"ДЛИНА ТРАССЫ 200 МЕТРОВ"
60 ? AT(3%,5%)"УРОВЕНЬ 1...5"
70 I$=INKEY$
80 IF I$="" then 70 ELSE KK%=VAL(I$)
90 IF KK%>=1% AND KK%<=5% THEN 130 ELSE ? STRING$(7%,7%)
   "НАЖИМАЙ КЛАВИШИ ОТ '1' ДО '5'"
120 GOTO 60
130 CLS
140 I%=10%
150 J%=0%
160 KK%=200%*KK%
170 POKE &0177660,&0100
180 IF INP(&0177716,&0100)=&00 THEN A%=PEEK(&0177662) ELSE 220
190 IF A%=8% THEN I%=I%-1% ELSE IF A%=25% THEN I%=I%+1%
200 IF I%<0% THEN I%=0% ELSE IF I%>30% THEN I%=30%
220 IF POINT(I%*8%+4%,104%)=1% THEN 340
230 J%=J%+1%
240 ? AT(I%,10%) A$
260 FOR K%=1% TO J% STEP 200%
270 ? AT(RND(K%)*31%,23%) B$;
280 NEXT
290 ?
300 IF KK%-J%<=0% THEN 370
310 FOR K%=0% TO KK%-J%
320 NEXT
330 GOTO 180
340 ? STRING$(20%,7%) AT(3%,10%) "ВЫ ПРОЕХАЛИ":FIX(J%*200/KK%);"M"
360 GOTO 380
370 ? AT(2%,10%)"******** Ф И Н И Ш ********"
380 POKE &0177660,&0100
390 GOTO 50
```

Заключение рецензента

Игровая программа, предложенная В.В.Ермаковым, будет интересна многим читателям. К числу ее достоинств относятся:

- широкий набор используемых операторов и функций БЕЙСИКа;
- простая логика программы;
- небольшое количество используемых переменных и лаконичность текста;
- достаточно быстрая реакция на действия игрока за счет использования операций с системными регистрами;
- постепенно изменяемая в процессе одной попытки сложность игры;
- возможность самостоятельного наращивания и совершенствования игры.

Вместе с тем описание алгоритма, которое сделал автор, сверхлаконично. Поэтому поясним работу программы.

В строках 20, 30 формируются переменные, содержащие символы, изображающие "лыжника" (код 182) и препятствие (код № 127) соответственно. Подсказка, ввод уровня сложности игры и контроль ошибки ввода обеспечивается в строках 50 - 120.

Строки 130 — 160 выполняют очистку экрана и начальную установку координаты "лыжника" I%, пройденного пути J% и длины трассы KK%. Оператор 170 маскирует прерывание от клавиатуры путем установки в единицу шестого бита регистра состояния клавиатуры с адресом 177660.

Строка 180 обеспечивает считывание кода нажатой клавиши из регистра данных клавиатуры (адрес 177662) в переменную A%. Клавиша нажата, если шестой разряд регистра управления системными внешними устройствами (адрес 177716) содержит 0.

Строки 190 — 210 в зависимости от содержимого переменной A% вычисляют новое значение координаты I% "лыжника", контролируя недопущения выхода за границы экрана. Факт наезда на препятствие устанавливается оператором 220. Пройденный "лыжником" путь увеличивается в строке 230. Строки 250 — 290 выводят в последнюю строку экрана препятствия, координаты которых определяются с помощью датчика случайных чисел RND. Количество препятствий в строке определяется длиной пройденной дистанции J%. Последний оператор в этой группе строк продвигает содержимое экрана на одну строку вверх.

Оператор 300 проверяет условие окончания трассы. Строки 310 — 320 создают задержку, зависящую от пройденного "лыжником" пути и длины трассы: чем меньше осталось до финиша — тем меньше задержка и, следовательно, выше скорость спуска.

Строки 340, 350, 370 разделяют два случая завершения спуска: успешного и неуспешного. Разрешение на прерывание от клавиатуры обеспечивается оператором 380.

В программе имеется неточность, внесенная, очевидно, при переписывании текста с экрана. В строке необходима переменная цикла, отличная от I%. Лучше всего I% заменить на K%.

Л.Н.Жариков

ЕСТЬ РАЗГОВОР

К вам обращаются оба редактора, готовящие выпуски ВТ и ее применение: Васильев Борис Михайлович — старший научный редактор и Васильева Наталья Алексеевна — младший редактор и оператор ЭВМ. Все наши выпуски оформляются ею в издательской системе Xerox Ventura Publisher и текстовом редакторе WORD-4.

И оба рецензента-энтузиаста — Л.Н.Жариков и В.А.Хмелюк.

Сразу приносим извинения всем нашим читателям, которые не дождались от нас ответа. Увы, это невозможно даже физически. Однако уверяем вас, что ни одно письмо не остается без внимания. К письмам читателей относимся очень серьезно, даже к сердитым, полностью критическим, пустым и невыполнимым по запросам. С ваших писем и началось формирование структуры выпусков ВТ. В 1987 г. и подсмотреть-то было некуда, мы оказались одними из первых. А рубрика "БК за рога" "живет" почти полностью на интеллекте наших читателей.

Просим извинения у тех, кому мы не выслали экземпляров, пропавших где-то в недрах почты "СОЮЗПЕЧАТИ". Это наша общая боль. Мы специально покупаем до 20 экз. в месяц (больше нам не дают), которые и уходят подписчикам на устранение почтового и полиграфического брака.

Уважаемые программисты! Ваши письма с предложениями и рекомендациями относятся к самым интересным и ожидаемым. К вам особое наше обращение.

Тем, кто пишет в журнал впервые. Статью нужно начинать с объяснения, для кого и для чего она написана. При описании программ нужно указывать, чем они отличаются от ранее опубликованных, описывать их особенности. Не бойтесь приводить подробности и формулы. Важно, чтобы материал был понятен не только вам, но и тем, кто с ним будет работать, а излишние подробности редакция уберет.

Если есть возможность, то лучше отпечатать статью на машинке, а если нет, то ее нужно аккуратно написать от руки. Иллюстрации и таблицы выполняются каждая на отдельном листе. Это же относится к программам. Программы для микрокалькуляторов следует писать в строку, без кодов, в тех обозначениях, которые приняты в № 5/91. В одной строке не более 30 знаков (знаком считается буква, цифра, знак препинания и пробел). Отправляемые программы должны быть выверены и не содержать непонятных исправлений. Желательно, чтобы автор оставил у себя копию отправляемых материалов.

В первую очередь рассматриваются программы, представленные четкими листингами на белой бумаге, изготовленные на новой красящей ленте. Все, что написано рукой с экрана, как правило, содержит ошибки. Такие программы приходится перепечатывать — это дополнительные ошибки. Отсюда сердитые письма наших читателей. Проверить программы в редакции мы (оба редактора и оба рецензента) не имеем возможности — в редакции нет БК и, как вы понимаете, времени. Один из рецензентов в Ленинграде (это также о времени). Следовательно, от руки написанные программы или лежат в редакции очень долго, или идут в печать (но при этом от брака избавиться не удается).

Программистов (и других читателей), обращающихся в редакцию, просим сообщать полностью имя, фамилия, отчество, адрес (с индексом), паспортные данные (для выплаты гонорара) с указанием наличия детей, телефон (служебный, домашний), год рождения. Просим также предоставлять нам (письменно) право помещать в конце вашей статьи данные об авторе (телефон, адрес или какие-то другие сведения по вашей подсказке).

Всем тем, кто желает разместить в журнале рекламу, сообщаем:

- цена рекламы с 1992 г. исчисляется из расчета 2000 руб за одну полную страницу. Для этого в редакцию нужно прислать гарантийное письмо и проверенный (вычитанный) текст, можно с фотографиями, чертежами и т.п.;
- цена рекламы для частных лиц — из расчета 500 руб. за одну полную страницу. При этом деньги перечислить предварительно (Издательство "Знание", Бауманское отд.Жилсоцбанка в г.Москве, код 20135, счет 608321). Текст рекламы и графику — письмом.

ПОПРАВКИ К СТАТЬЯМ, ОПУБЛИКОВАННЫМ РАНЕЕ

"Вычислительная техника" № 2 за 1990 г. (Синтезатор речи). Перепутаны местами операторы STRING$ и CHR$ (стр.34, средний столбец, внизу). Текст должен выглядеть так:

1. В вильнюсской версии Бейсика (БК-0010.01) "стрекочущий" звук обычно получают с помощью печатаемой в цикле CHR$(7). Это минимум три оператора. Тот же результат можно получить одним оператором ?STRING$(N,7);...

Кроме того, хочу в дополнение сказать, что функция STRING может быть использована для замены отсутствующего в данной версии оператора SCROLL: ?STRING$(N,C) вызывает рулонное смещение содержимого экрана на № строк вверх (C равно 10, 27, 30, 31) или вниз (C равно 26, 28, 29). При этом служебная строка помех не создает и по экрану не перемещается, но при работе оператора может возникать неприятное мелькание, поэтому ее можно заранее погасить.

Недопечатано четыре строки текста программы "Синтезатор речи", содержащие выводимые на экран тексты подсказок. Чтобы вернуть программе работоспособность, предлагаем дополнение к уже введенному вами тексту, опубликованному в № 32:

160354	172742	020340	166353	173741	175751	000365	020351
167747	167767	164762	162764	027040	020351	166363	175765
165341	162764	027040	167744	171440	164767	160744	164756
020361	115441	115040	000000				

Контрольная сумма для всей программы равна 130277 (восьм.). Напомню еще раз, как ее проверить: $ 1000A1310ДХ=130277 (символы, выводимые БК, подчеркнуты).

И еще. При окончании работы программы производится выход в монитор БК (приглашение "?"). Если вы захотите воспользоваться созданным программой закодированным текстом речи, нужно вернуться в режим ТС. Описанный в руководствах по БК способ (выйти в ФОКАЛ и затем из него войти в ТС) непригоден — ФОКАЛ может затереть записанные в ОЗУ коды. Нужно воспользоваться следующим способом:

? <РУС> Т <ВВОД>
+ ТС
$

В программе "Текстовый редактор" ("Вычислительная техника" № 3 за 1991 год) на стр. 21 во второй строке блока с начальным адресом 2000 недопечатан код 146715, в результате чего искажается текст заставки-подсказки и все последующие контрольные суммы (хотя сама программа-редактор сохраняет работоспособность). Вместо напечатанного во второй строке указанного блока следует вводить:

020310 151320 143717 140722 146715 153440 014412 014431

В статье "Быстрый ввод операторов" — "Вычислительная техника и ее применение", № 12 за 1990 год, стр.35, были допущены некоторые неточности:

- в строке 50 программы вместо "-28789" следует читать "28789", вместо "28295" следует читать "-28295";
- при вводе программы необходимо особое внимание обратить на числа со знаком "минус" в операторах DATA: знак "минус" помещен в конце строки, а сами числа перенесены в начало следующей;
- в строке 60 программы вместо "STER" следует читать "STEP";
- далее по тексту — вместо BSAVE "SKEY:, 256,482 следует читать BSAVE "SKEY",256,482.

Приносим извинения нашему автору Евгению Петровичу Тарасенко, статья которого (см. № 5 за 1991 г., стр.37) по ошибке подписана А.Тарасенко.

В статье "Введение в программирование на языке Си" в выпуске "Язык программирования Си" (№ 4/1991) в таблице "Специальные символьные константы" (стр.37) в столбце "Символ" значения следует читать так: '\n', '\t', '\v', '\b', '\r', '\a', '\a\', '\', '\"', '\\', '\0'.

Соответственно в тексте вместо прямой наклонной черты (/) у этих символов надо читать обратную (\ ).

Прогресс неотвратим, и его проявления в сфере компьютеризации вслед за "комиссионками" полегоньку пролезают и на прилавки магазина "Электроника". Но пытливые пользователи с радиолюбительским стажем не могут ждать промышленных милостей. Продолжая начатую ранее в этой серии тему, рассмотрим еще кое-какие возможности домашнего дооснащения нашего бытового компьютера.

Н.Герман

Еще о рогах для БК

Действительно, прогресс, хочешь не хочешь, но, выражаясь словами классика, "свое действие оказывает". И вот в "Электронике" уже можно купить массу полезнейших компьютерных аксессуаров, вплоть до принтеров и дисководов, не говоря о "мыши" и кое-каких средствах сопряжения. Что же до злосчастнейших джойстиков, то их предлагают даже два типа, так что здесь имеется формально-рыночная возможность выбирать, тем более что один из них внешне крайне выгодно от другого отличается (речь идет о типах, поскольку, к какому роду следует относить само слово "джойстик", так полюбившееся автору, решить пока совершенно невозможно). Конечно, цены, установленные на эти изысканные продукты не без оглядки на те же "комиссионки", — вопрос особый... Взять хотя бы упоминавшийся нами в первой статье пишущий узел для принтеров, выпускаемый теперь и у нас в видах покрытия потребностей в расходуемых на печатание импортных чернилах. И все бы ничего, да только стоит эта одноразовая чернильница примерно с 2 мл чернил ни много ни мало — 25 р.! Ни писать, ни читать не захочется... Правда, наши неистребимые левши умудрились, говорят, перезаправлять подобные заморские "непроливашки" с помощью вполне многоразовых медицинских шприцев и самых что ни на есть отечественных чернил "Радуга"! Впрочем, ближе к делу!

Очередная самодеятельная конструкция, над которой мы предлагаем потрудиться мастерам паяльника, — счетчик. Создание любой электронной аппаратуры сейчас всегда начинается с выбора элементной базы. Поскольку порт БК рассчитан на подключение микросхем ТТЛ, да и обилие и доступность их ассортимента сейчас, пожалуй, максимальны, для счетчика мы советуем приспособить пока микросхемы именно серий ТТЛ. Удобнее всего нам будут двоичные четырехразрядные счетчики типа ИЕ7 (К155ИЕ7 или аналогичные из серий 133, 555 и т.д.). Соединив последовательно 4 таких счетчика, двоичные выходы которых можно непосредственно подключить ко всем 16 регистрам порта ввода БК, мы получим едва ли не простейшее, но достаточно полезное счетное устройство (рис.1). С его помощью

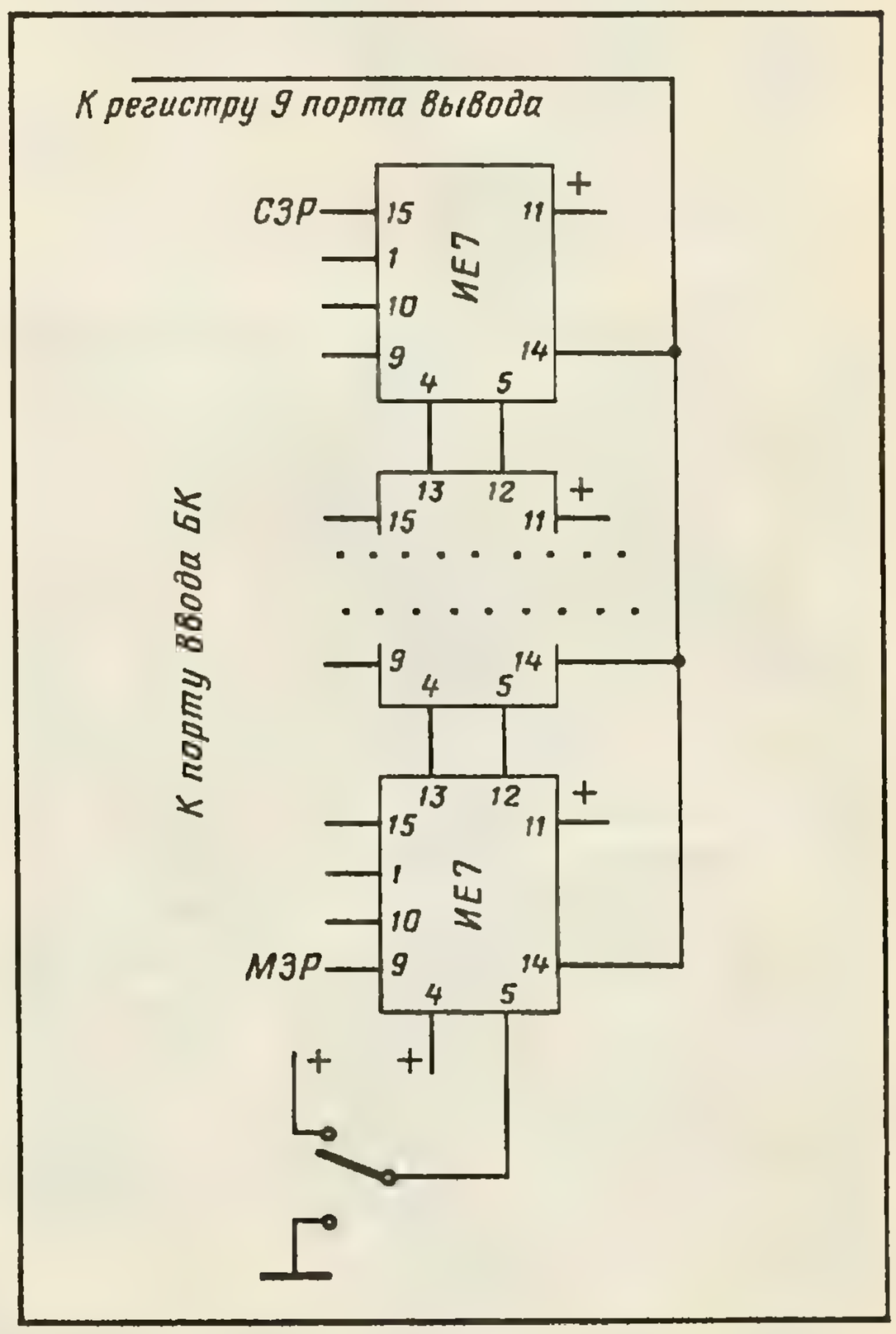


Рис.1

БК сможет считать все, что будет в состоянии создавать на счетном входе отчетливые импульсы (счетчик ИЕ7 увеличивает содержимое своих регистров на единицу на каждый положительный перепад 0 — 1 на счетном входе). Для начала естественно попытаться подсчитывать число замыканий какого-нибудь механического контакта — кнопки или тумблера (компьютеризация этого процесса пока достаточно нелепа, и это лишь упражнение в освоении нового устройства).

Организуем-ка программу для фиксации текущего содержимого счетчика и для демонстрации его на дисплее в цифровом, либо в графическом виде, либо, наконец, и в том и в другом:

Программы 1 (BASIC и ФОКАЛ)

```
10 C=32768-INR(-52,-1)
20 ? AT(0,0) B,C
30 PSET (B,230-C)
40 B=B+1
50 IF B2>56 THEN B=O
60 GOTO 10
1.1 X FP(1,2000)
1.2 X FP(2,2000)
2.1 S C=65535+FP(0,100000)-FP (0,77777)
2.2 X FK(0,0); T B,C
2.3 X FT(1,B,230-C)
2.4 S B=B+1
2.5 IF (B-256)1.2
2.6 S B=O; G
```

Обратите внимание на способ вычислений показания счетчика, необходимость которых связана с особенностями представления чисел в регистре порта (в частности, старшего — знакового разряда).

Попытавшись несколько раз нажать на кнопку счета, с помощью этих программ нетрудно будет обнаружить, что после каждого нажатия содержимое счетчика увеличивается вовсе не на единицу, как того хотелось бы, но на какое-то хаотически меняющееся количество (обычно в пределах 3 — 15 единиц). Происходит это, как выражаются теперь специалисты, из-за "дребезга" контактов. Сходные неприятности могут наблюдаться, однако, и при подсчете импульсов от какого-либо чисто электронного устройства, где причин для механического дребезжания быть не должно.

Не вдаваясь в дополнительные подробности, укажем лишь, что для исключения подобных явлений придется на входе счетчика соорудить дополнительный формирователь, обеспечивающий недвусмысленное однократное срабатывание первого регистра ИЕ7. Схемы таких формирователей совершенно стандартны[1], и нам может потребоваться еще одна микросхема с так называемыми триггерами Шмитта (К155ТЛ1 и т.п.).

Однако, имея в виду дальнейшие возможности подключения к счетчику каких-либо иных электронных устройств, например, часов, работающих обычно на микросхемах МОП, нам будет иметь смысл освоить в формирователе схемы из серий КМОП — 176, 561 или 564. Весьма удобной здесь для нас оказывается микросхема К561ЛН1. Во-первых, к ее выходам можно подключать и схемы ТТЛ (по нормам — не более двух сразу), тогда как рядовые схемы КМОП не в состоянии раскачивать входы ТТЛ. Выходы ЛН1, далее, можно переводить в так называемое третье, выключенное, состояние и поэтому на основе этой схемы можно строить удобные переключатели источников многоразрядной информации.

После того как наш счетчик с формирователем начнет уверенно отсчитывать положенные единички, должен возникнуть естественный вопрос: что же нам теперь эдакое посчитать? Вот если, к примеру, в вашем распоряжении имеется счетчик Гейгера (который сейчас многие мечтают приобрести, наивно полагая, что сумеют тем самым обезопасить свое существование), можно будет заняться даже такими капитальными проблемами, как исследование вариаций космических лучей!

Более реальная и интересная возможность — организовать счет секунд мирового времени. Если в вашем домашнем хозяйстве пока нет готовых электронных часов, то советуем приобрести хотя бы детский конструктор — набор деталей для изготовления полезного бытового прибора (в продаже бывают наборы на НМОП микросхеме К145ИК1901 или же на КМОП микросхемах серии 176, которые для наших изысканий более удобны). Если имеется возможность приобрести часовой кварц (в продаже встречаются кварцы на 32768 Гц), то с дополнительной микросхемой

1 *См.: Шило В.Л. Популярные цифровые микросхемы. — М.: Радио и связь, 1987. Также см.: Вениаминов В.И. и др. Микросхемы и их применение. — М.: Радио и связь, 1989.*

К176ИЕ5 (либо ИЕ12) можно самому собрать "сердце" таких часов—высокостабильный кварцевый генератор.

При попытке подключения часов или иного электронного устройства ко входу нашего счетчика может возникнуть очередная проблема — "сопряжение входов-выходов", особенно в тех случаях, когда подключаемое устройство питается от своего источника, да еще и с иным напряжением питания. Можно было бы соорудить еще какой-нибудь универсальный входной усилитель-формирователь, покрывающий большинство мыслимых потребностей. Но для начала следует воспользоваться более простой схемой действий: замкнув вход нашего формирователя страховочным сопротивлением (а для дополнительной острастки параллельно такому сопротивлению в 20 кОм можно подключить еще и стабилитрон — КС147 или КС156), подавать на вход исследуемые импульсы через небольшой конденсатор. Емкость его следует подобрать опытным путем, постепенно увеличивая ее до тех пор, пока импульсы не начнут уверенно засчитываться (после чего для верности эту емкость можно увеличить еще примерно вдвое). Таким путем мы сможем для каждого источника сигналов подобрать свой входной конденсатор.

Имея теперь источник точно калиброванных секундных импульсов, можно немедленно соорудить и неплохой частомер. В обычных частомерах наибольшие схемные сложности возникают при организации циклических или однократных отсчетов в заданных интервалах и индикации результатов.

Нам же не грех поручить всю эту "оргработу" нашему премудрому компьютеру—он и не с такими задачами в состоянии справиться! Единственное, что мы потребуем от остальной схемы, — создание периодических "пачек" исследуемых импульсов точно определенной продолжительности, например по 0,5 с (у схемы К176ИЕ5 имеются выходы, откуда можно извлекать частоты 1, 2, 64 Гц, а также исходную частоту 32768 Гц).

Если подать "меандр" частотой 1 Гц на "запрещающий" вход микросхемы К561ЛН1, наш формирователь будет периодически включаться и выключаться, образуя нужные нам серии. В

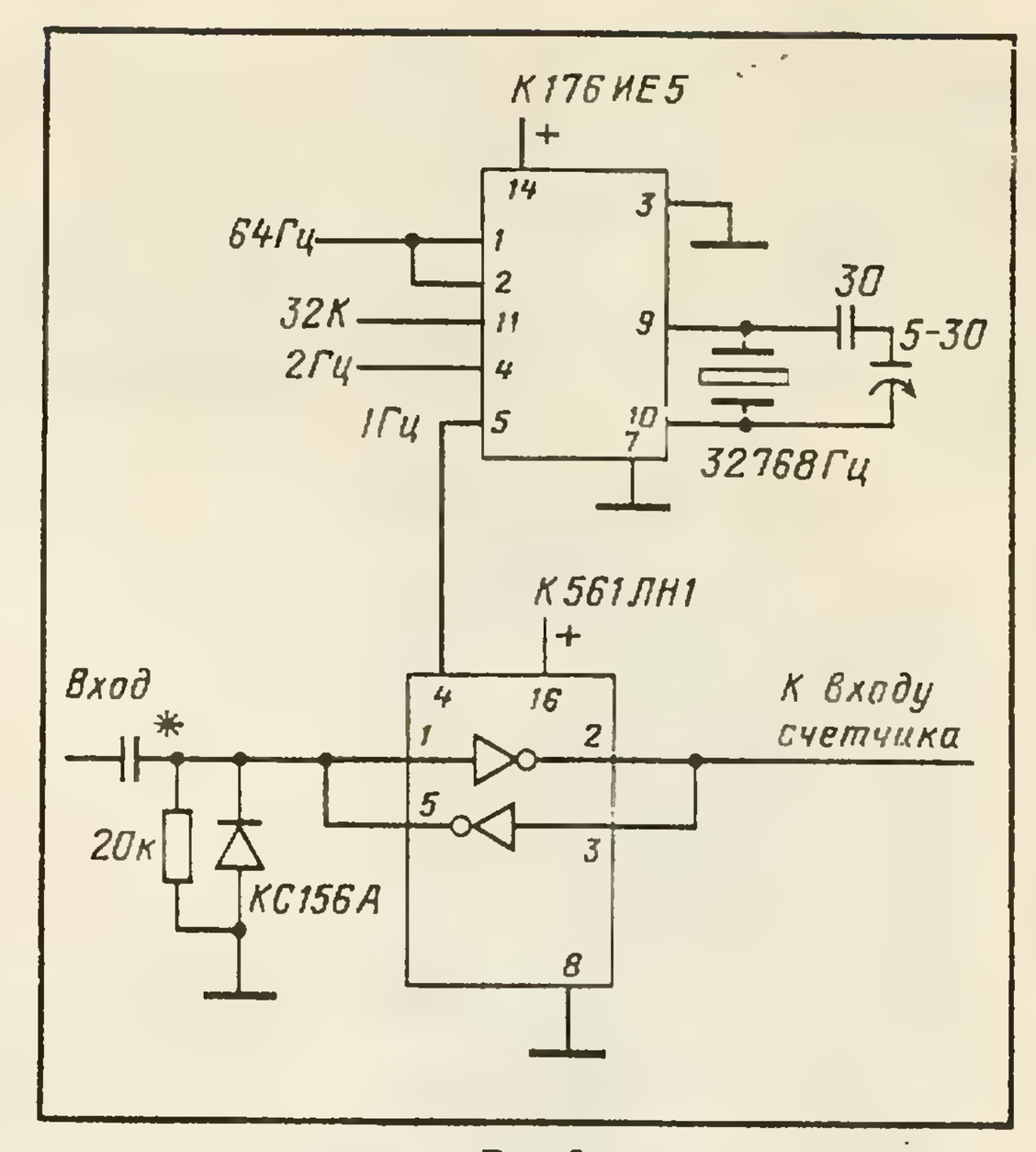


Рис.2

разрывах этих серий после считывания результата мы заставим также БК производить и необходимое "обнуление" всех регистров счетчиков. Для этого мы подсоединим все входы сброса (R) счетчиков к одному из регистров порта вывода. Пусть это будет 10-й регистр (первые десять, начиная с нулевого и заканчивая номером девять, нам еще понадобятся).

Окончательная схема частомера изображена на рис.2. Впрочем, ее можно дополнительно еще долго совершенствовать, и поэтому ее стоит собрать пока на какой-нибудь макетной плате.

Полное потребление подобной схемы при использовании 4 счетчиков К133ИЕ7 у автора не превышало 160 мА, так что ее можно было бы питать непосредственно от источника питания самого БК. Однако вопреки тому, что написано по этому поводу в книгах[1], в порту моего БК отведенные под 5 В источника контакты не были вовсе к нему присоединены.

В подобной ситуации, если у вас нет пока другого источника, можно соорудить для БК дополнительный переходник из стандартных разъемов СШ-5. Но если уж браться за серьезное оснащение нашего компьютера, то завести дополнительный источник все равно не миновать, причем для дальнейшего следует предусмотреть и получение напряжений в ± 15 В (при

1 *См. Осетинский Л.Г. и др. Фокал для микро- и мини-компьютеров. — Л.: Машиностроение, 1988.*

общей мощности порядка 200 Вт).На первых же порах для налаживания нашей схемы можно будет обойтись обычной "плоской" батарейкой на 4,5 В.

Теперь нам осталось научить БК опознавать появление пачки исследуемых импульсов с момента окончания счета:

Программы 2

```
100 ' COUNTER
110 ? AT(0,0) C
120 GOSUB 200
130 GOTO 110
200 POKE-52,0
210 POKE-52,1024
220 GOSUB 300
230 IF C >< 0 THEN 200 ELSE K=1.5
240 GOSUB 300
250 IF C=O THEN 240
260 IF C=K THEN RETURN ELSE K=C
270 GOSUB 300
280 GOTO 260
300 C=32768-INR(-52,-1)
310 RETURN
1.1 X FK(0,0);T C
1.2 D 2;G 1.1
2.1 X FP(1,2000);X FP(2,2000)
2.2 D3
2.3 I (C)2.1,2.4,2.1
2.4 S K=1.5;D 3
2.5 I (C)2.6,2.4,2.6
2.6 I (C-K)2.7,2.8,2.7
2.7 S K=C;D 3; G 2.6
2.8 R
3.1 S C=32768-FP(0,177777)
```

Следует иметь в виду, что при измерении достаточно низких частот, в особенности при работе с более быстрым БЕЙСИКом, может случиться так, что за очередной цикл опроса содержимое счетчиков не успевает измениться, и тогда БК не сумеет в этой путанице разобраться. В этих случаях можно ввести в наши программы замедляющие циклы:

Программы 2а

```
236 GOSUB 350
266 GOSUB 350
350 FOR G=O TO 50
360 NEXT G
370 RETURN
2.4 D 4; S K=1.5;D 3
2.7 D 4; S K=C;D 3; G 2.6
4.1 F G=O,50
```

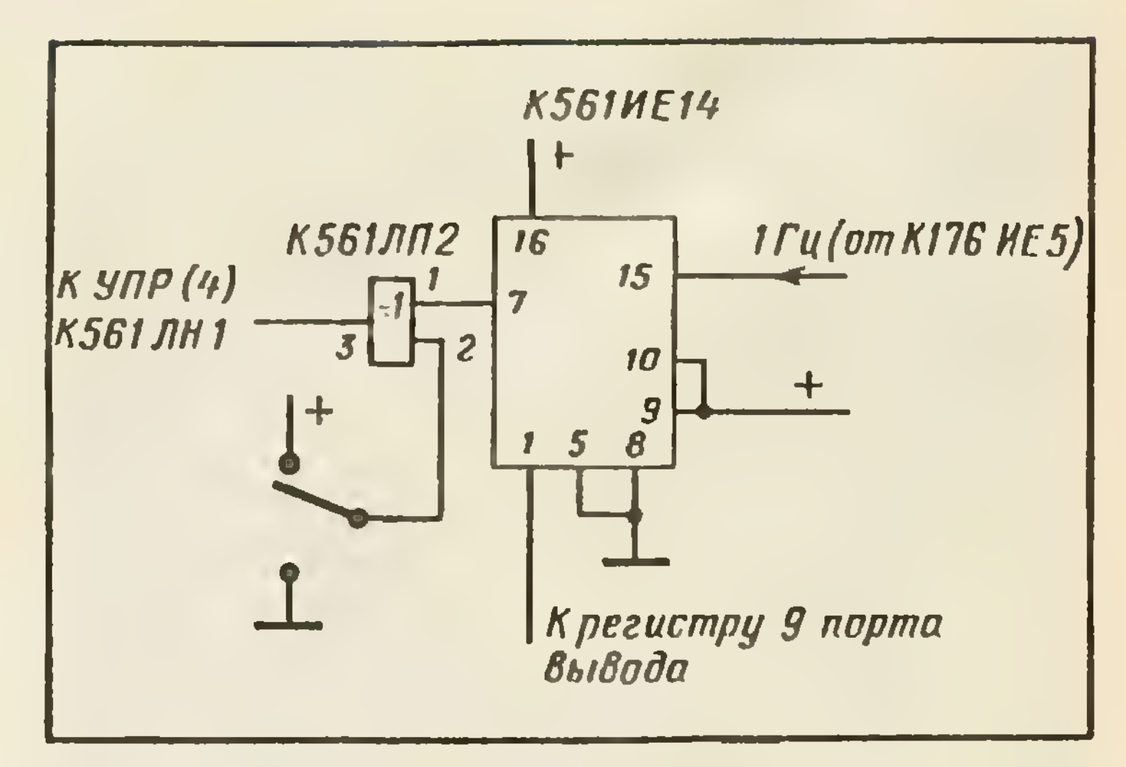


Рис.3

Вообще же радиолюбителям частоты ниже 10 Гц довольно редко бывают необходимы. Да и считать такие частоты на полусекундных интервалах, причем с помощью компьютера, не совсем рационально. Но если подобная необходимость возникает либо также если нам потребуется отсчитывать максимальное число счетов, можно с помощью дополнительной микросхемы просто расширить счетный интервал (из схемы К176ИЕ12 можно непосредственно извлечь минутную частоту—1/60 Гц). Возможный вариант расширения счета до 15 с приведен на рис.3.

С помощью такого усовершенствованного частотомера можно даже взяться за довольно хлопотную задачу, требующую массы времени и настойчивости, — исследование относительной стабильности хода пары электронных часов, скажем готовых промышленных бытовых часов (средняя частота которых обычно известна очень хорошо по их уходу от сигналов точного времени) и изготовленных нами для нашего частотомера. Подстроив с помощью дополнительных емкостей частоты обоих часов, можно начать систематические исследования влияния на рабочую частоту двух самых главных дестабилизирующих факторов—напряжения питания и температуры. Для контролируемого воздействия температуры на отдельные элементы схем достаточно удобно пользоваться обычным бытовым феном, направляя струйку теплого воздуха на обследуемый элемент. Результатом подобного самодеятельного научного исследования в принципе может явиться заметное улучшение стабильности хода обоих ваших часов.

Если вы, уважаемые исследователи, рискнули воспроизвести также

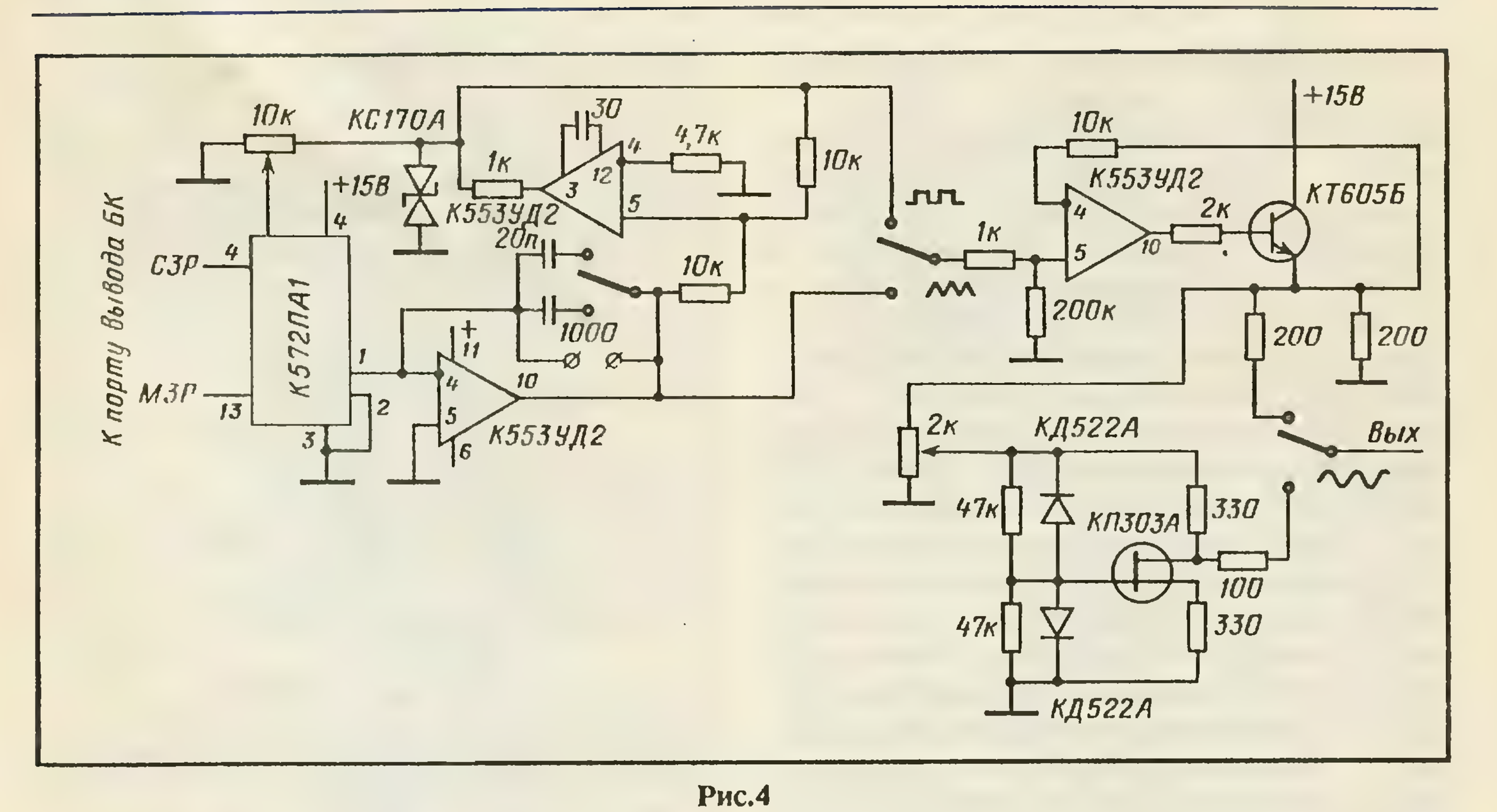


Рис.4

предлагавшийся нами в предыдущей статье музыкальный генератор (см.: № 11 настоящей серии за 1989 г.), то очередной задачей для БК-частотомера должна, несомненно, явиться точная калибровка частот этого генератора (на рис.4 мы приводим его более полную и совершенную схему). Хотя его частота, как и у всякого генератора, подвержена определенным уходам, для большинства целей оказывается достаточным не измерять ее всякий раз, но вычислить по заданному цифровому коду.

Калибровочная зависимость оказывается предельно простой для диапазона низких частот (не выше 2 — 3 кГц), где хорошо соблюдается линейная связь частоты и кода. В диапазоне же до необходимых нам 20 кГц становится заметной определенная нелинейность (здесь удается аппроксимировать реальную зависимость полиномом 3-й степени).

Заметим, кстати, что подобная калибровка представляет собой вполне точный способ контроля качества используемой в генераторе микросхемы ЦАП—К572ПА1 и ею можно воспользоваться для выбора наилучшего экземпляра (для ряда применений требования к этой схеме могут быть куда как менее строгими. Сейчас, например, ее нередко используют для цифрового управления громкостью. Ясно, что градация громкости с точностью даже в 1%—излишество).

Программы 3

```
100 ' LINEARITY TEST
110 GOSUB 200
120 PSET(A/4,190-C/50) '  или с подбором B и D
120 PSET(A/4,100+(C-A*D)*B)
130 A=A+1
140 IF A >1024 THEN STOP ELSE 110
200 POKE-52,0
210 POKE-52,2048-A
1.1 D 2;X FK(0,0);T A,C
1.2 X FT(1,A/4,230-C/50)
1.3 S A=A+1
1.4 I (A-1024)1.1
1.5 S A=O;G
2.1 X FP(1,2000)
2.2 X FP(2,4000-A)
```

Как мы уже указывали в предыдущей статье, подобный генератор, управляемый компьютером, весьма удобен для всевозможных измерений, которые нередко приходится проводить заядлым радиолюбителям, например, для снятия частотных характеристик узлов звуковоспроизводящей аппаратуры. Но согласитесь, что неплохо было бы заставить БК взяться и за вторую часть задачи, т.е. не только устанавливать нужную частоту, но и производить сами измерения напряжений. Для этого понадобится еще и АЦП—аналого-цифровой преобразователь.

Здесь автор не может не похвалиться, что ему в свое время удалось купить все в той же "Электронике" готовую и

вполне работоспособную (сейчас говорят — рабочую) микросхему АЦП — КР572ПВ1. Теперь же, вероятно в видах новых экономических установлений, предпринимаются попытки прекратить розничную торговлю микросхемами и другими необходимыми компонентами в "Электронике" (зато там появились импортные микросхемы по макроценам!). При этом волна продажи уцененных компонентов в соответствующих магазинах еще не докатилась до АЦП (а вот ЦАП К572ПА1 там можно уже разыскать). Но ведь у всех магазинов, где должны были бы продаваться столь нужные радиодетали, всегда можно воспользоваться нашим родным способом — обратиться к толпе пусть порою уже пьяненьких спекулянтов (впрочем, по уголовному амплуа это скорее "барыги", так как торгуют они преимущественно не скупленным, а краденым). Если нужного товара нет сегодня, его принесут в удобное время, причем вам может быть выдана даже устная гарантия качества продукта! Впрочем, вопрос о торговле радиодеталями — отдельная тяжелая тема...

Итак, попытаемся все же ориентироваться на то, что АЦП ПВ1 читателю удалось так или иначе достать. В противном случае мы можем порекомендовать уже упоминавшуюся книгу Р.Фелпса "750 практических электронных схем" (М.: Мир, 1986). Видимо, к моменту выхода книги в США (1983 г.) микросхемы АЦП еще не получили достаточно широкого распространения, и поэтому в книге содержится не менее шести вариантов схем АЦП с использованием микросхем ЦАП, в том числе типа К572ПА1-DAC7520 и т.д., и так называемых регистров последовательного приближения (РПП типа К155ИР17).

Стоит указать, что АЦП К572ПВ1 представляет собой именно комбинацию ЦАП, сходного с К572ПА1 и РПП. Так что синтез АЦП из более доступных компонент — дело вполне реальное. Возможен также еще более капитальный синтез АЦП с некоторым снижением точности — см., например, упоминавшуюся уже книгу В. Шило, с. 219.

Хотя микросхема ПВ1 объединяет массу сложных и разнородных узлов, для построения полной схемы АЦП потребуется (помимо источника питания на : В и + 15 В) еще как минимум два дополнительных элемента — компаратор и источник опорного напряжения (рис.5).

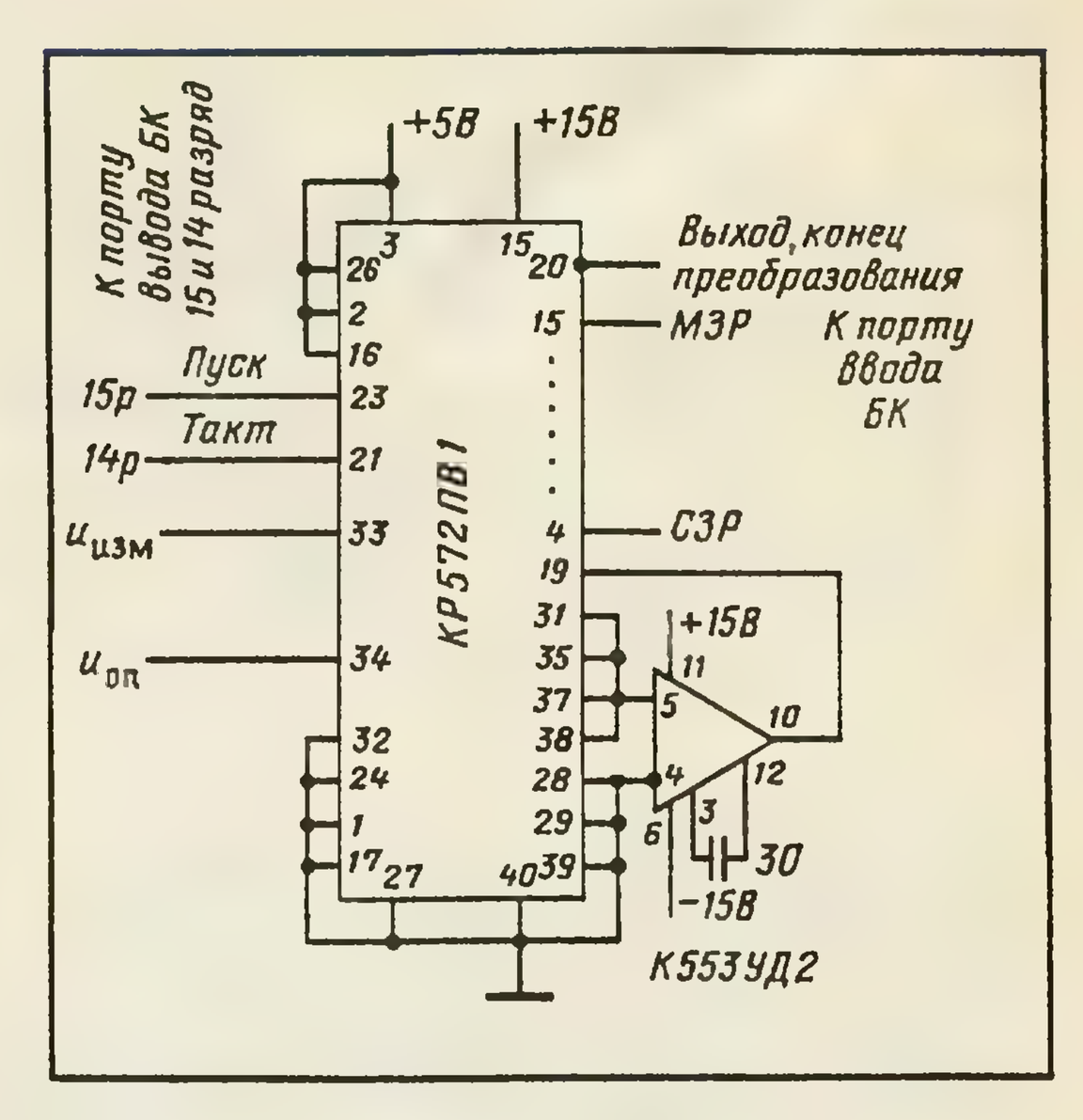


Рис.5

Правда, в качестве компаратора нам пока подойдет практически любой операционный усилитель, а при отладке схемы вместо ИОН допустимо использовать хотя бы напряжение от самого источника питания. Итак, снова за паяльник!

В числе необходимых для работы нашей схемы устройств мы не указали еще тактовый генератор, поскольку можно и этот род деятельности поручить неутомимому БК. Программы, предназначенные для измерения напряжения одной полярности, могут выглядеть следующим образом:

Программа 4

```
50 GOSUB 100
60 ?AT(0,0) X
70 GOTO 50
100 POKE-52,16383
110 POKE-52,-1
120 FOR K=O TO 24
130 POKE-52,-16385
140 POKE-52,-1
150 NEXT K
160 X=4095-INP(-52,4095)
170 RETURN
1.1 D 2;X FK(0,0);T X;G
2.1 X FP(1,140000)
2.2 X FP(2,140000)
2.3 F K=0,24;D 3
2.4 S X=4095-FP(0,7777)
3.1 X FP(1,40000)
3.2 X FP(2,40000)
```

При использовании же внешнего тактового генератора (например, генератора часов на 32768 Гц) строки 120-150 или 2,1-2,2 можно исключить. Формально при этом следовало бы убеждаться, что цикл преобразования завершился, и лишь потом производить считывание результата (при этом на выходе АЦП "конец преобразования" возникает лог.1, удерживающаяся до подачи очередной команды "пуск").

Однако при использовании указанной тактовой частоты даже наш быстрый БЕЙСИК, не говоря уже о ФОКАЛе, не успевает за всеми циклами АЦП, и поэтому такая проверка оказывается излишней.

Микросхема К572ПВ1 по паспортным данным обязана работать до максимальной тактовой частоты 250 кГц, так что темп опроса ее может достигать 10 кГц. Для БК (по крайней мере с нашими программными средствами) это слишком резвый темп, поскольку даже при работе с внешним тактовым генератором он еле-еле успевает проделать десяток измерений в секунду. Тем не менее для записи не слишком быстрых процессов весь наш агрегат можно использовать в качестве своеобразного шлейф-осциллографа:

Программа 5

```
50 POKE-52,32767
60 POKE-52,-1
70 X=4095-INP(-52,4095)
100 PSET(B,230-X/20)
110 B=B+1
120 IF B > 256 THEN B=O
130 GOTO 50
1.1 X FP(1,100000)
1.2 X FP(2,100000)
1.3 S X=4095-FP(0,7777)
1.4 X FT(1,B,230-X/20)
1.5 S B=B+1
1.6 I (B-256)1.1
1.7 S B=O;G
```

Имеется возможность еще несколько поднять темп работы нашего устройства, если после каждого измерения не сразу строить результат на дисплее, но запоминать его до окончательного накопления данных всех измерений (256 или 512 точек на весь дисплей):

Программа 5а (только БЕЙСИК)

```
70 POKE12300+B,PEEK(-52)
130 FOR M=12300 TO 12556
140 X=INP(M,4095)-4095
150 PSET(12300-M,230+X/20)
160 NEXT M
120 IF B < 256 THEN 50
100 исключить.
```

Здесь стоит заметить, что выбор ячеек памяти, куда производится запись результатов, — дело достаточно тонкое, поскольку существует опасность попадания в ячейки, уже занятые содержимым каких-либо программ, что может приводить к самым разнообразным чудесам (вплоть до необходимости перезапуска процессора, т.е. и стирания всего содержимого памяти вместе со всеми имеющимися там программами). При не слишком загруженной памяти выбор номера ячейки начала массива данных (12300) позволял на БК автора записывать в последующие ячейки чуть более 3 тыс. данных (заметим, что число, считываемое из порта, трактуется БК как целое, и для его записи (в БЕЙСИКе!) может использоваться всего одна ячейка).

У нас теперь вырисовывается возможность производить запись самых медленных колебаний (единицы Герц), в том числе создаваемых нашим генератором (чтобы воочию рассмотреть их форму). Для этого осталось преодолеть лишь одно затруднение: нужно сделать исследуемый сигнал строго однополярным. Но вообще-то, схема ПВ1 вполне приспособлена и для измерений двуполярных напряжений, хотя это требует дальнейших усложнений. Интересно, что в специальной книге Б.Г.Федоркова и В.А.Тельца "Микросхемы ЦАП и АЦП" (М.: Энергоатомиздат, 1990) этот немаловажный момент упоминается лишь вскользь. Во-первых, теперь нам потребуется еще один ИОН (противоположной полярности), выход которого необходимо подключать к АЦП вместо прежнего источника в моменты перехода регистрируемого напряжения через нуль. Для определения этого момента понадобится, следовательно, еще один компаратор.

Но оказывается, что в эти моменты необходимо менять и смысл сигнала сравнения для АЦП, что можно сделать как минимум тремя способами: переключением входов основного компаратора (это довольно грубый прием), подключением

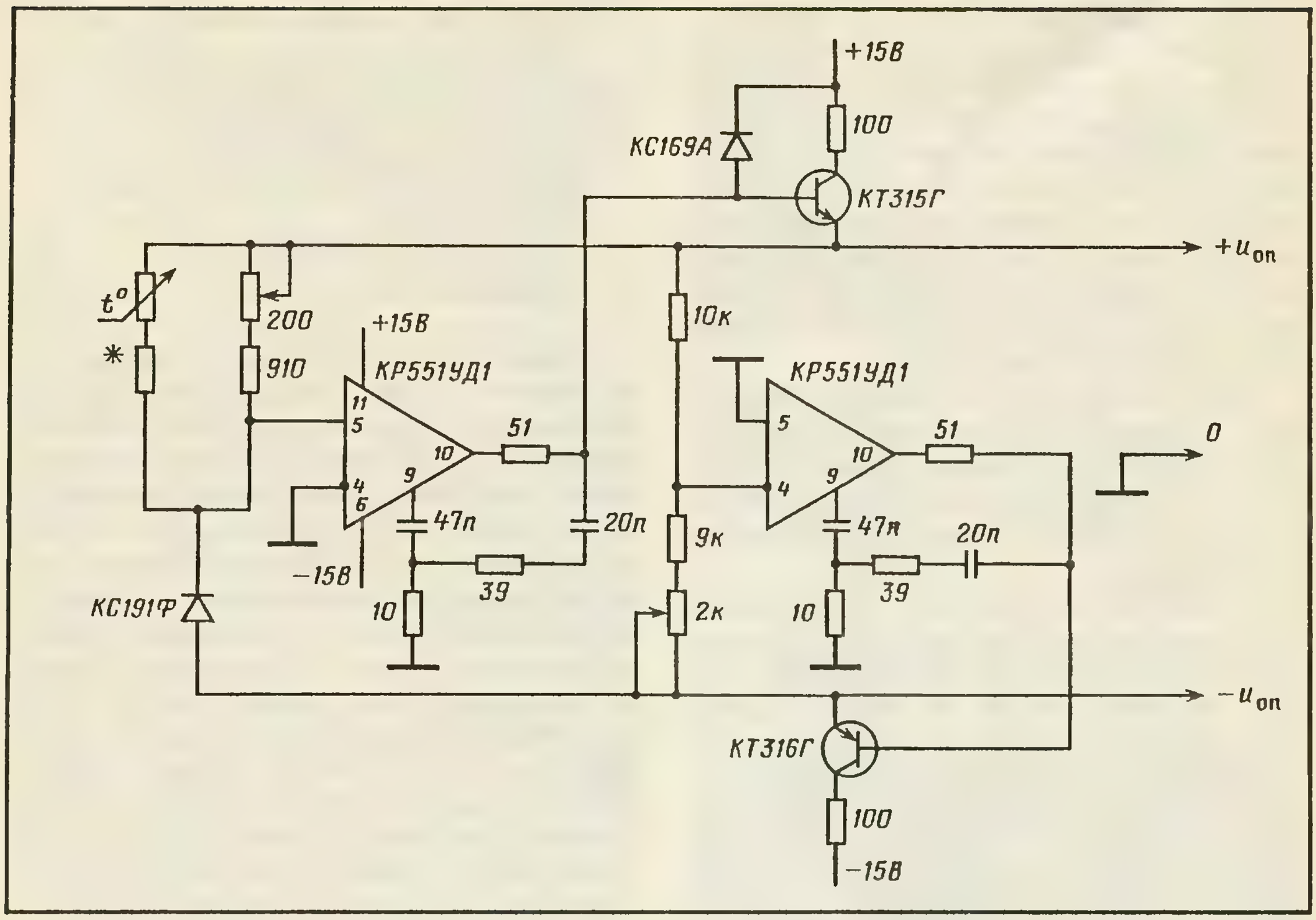


Рис.6

параллельно ему еще одного "перевернутого" компаратора с переключением теперь уже выходов этих двух компараторов (автор ухитрялся производить эти переключения на начальном этапе освоения двуполярных сигналов просто вручную!) и, наконец, включением дополнительного управляемого инвертора (элемента исключающее ИЛИ, типа К176ЛП2) в цепь сигнала сравнения.

При этом следует только учесть то, что амплитуда сигнала сравнения должна быть больше 10 В, так что используемая здесь микросхема должна такое напряжение обеспечивать, а также то, что отрицательные сигналы на вход КМОП микросхем подавать нельзя (придется предусмотреть здесь охранный диод). Оставим, впрочем, это направление для самостоятельной разработки нашим опытным радиолюбителям.

Мы можем наметить здесь еще несколько возможных направлений совершенствования набора наших компьютерных "безделушек". Радиолюбителям чаще приходится измерять амплитуды переменных напряжений, и поэтому следует оснастить наш АЦП детектирующим устройством. На первых порах будет достаточно простейшего пикового диодного детектора, но в дальнейшем неплохо было бы собрать детектор средневыпрямленного значения (на ОУ с хорошим быстродействием), а затем, может быть, детектор эффективного напряжения (это значительно сложнее). Следует подумать и об измерении достаточно слабых сигналов и малых э.д.с. (например, от термопар), для чего стоит разработать надежный усилитель (в том числе и усилитель постоянного тока). Возможные варианты этих устройств читатель найдет в упоминавшихся нами книгах, в дополнение к которым мы можем рекомендовать еще руководство Н.В.Воробьева и В.Д.Вернера "Элементная база и схемотехника средств сопряжений" из серии "Микропроцессоры" (М.: Высшая школа, 1984), а также книгу С.Д.Бирюкова "Цифровые устройства на МОП интегральных микросхемах" (М.: Радио и связь, 1990).

Потенциальная точность преобразования нашего АЦП в принципе должна соответствовать погрешности в один счетный разряд, т.е. 0,025% (для схем ПВ1А). Достичь реально подобной точности непросто, но даже погрешность в 0,1% может считаться вполне приличной (стре-

лочных приборов такого класса почти и не бывает). Один из первых шагов к повышению точности АЦП — создание качественного двуполярного ИОН. Основу современных ИОН составляют прецизионные стабилитроны (например, КС191Ф). Стабилитрон обычно включается в плечо моста, для питания которого используется вырабатываемое ИОН опорное напряжение (см. рис. 6). Поскольку сопротивление стабилитрона резко нелинейно, баланс моста, обеспечиваемый ОУ, возможен только при одном значении напряжения (одной полярности!). Второй ОУ обеспечивает равенство абсолютных значений выходных напряжений. В принципе баланс моста возможен и при "перевороте" всей схемы—при прямом включении стабилитрона (рабочее его включение — "обратное"). Чтобы такого переворота не происходило, на базу одного из триодов через дополнительный стабилитрон подается (нелинейное) смещение. В плечи моста можно включить дополнительные термозависимые элементы—термисторы, диоды и т.д. — для снижения влияния температурных дрейфов элементов схемы. С этой же целью в качестве ОУ следует использовать схемы с малым напряжением смещения и с малым его дрейфом, например К140УД17 или К153УД5, К551УД1.

Налаживание этого ИОН и подбор термокомпенсационных элементов может потребовать длительных и кропотливых измерений при контролируемых тепловых воздействиях на отдельные элементы (здесь снова можно использовать бытовой фен).

Для этих канительных процедур как раз стоит приспособить наш БК-АЦП. В качестве же опорного источника при этом можно выбрать свежую гальваническую батарею (типа "Крона"), промерянную на хорошем цифровом приборе. Важно только проследить, чтобы эта батарея не подвергалась ненужным нагружениям, в особенности случайным коротким замыканиям, после которых ее напряжение может долго "ползти". Следует учитывать, что изображенная нами схема АЦП имеет довольно низкое входное сопротивление, поскольку измеряемое напряжение подключается к АЦП через внутренние резисторные делители. В более совершенных вариантах, которые можно найти в рекомендованной литературе, сравниваемые напряжения подключаются через буферные ОУ.

Мы видим, что у радиолюбителей имеется неограниченное поле деятельности по

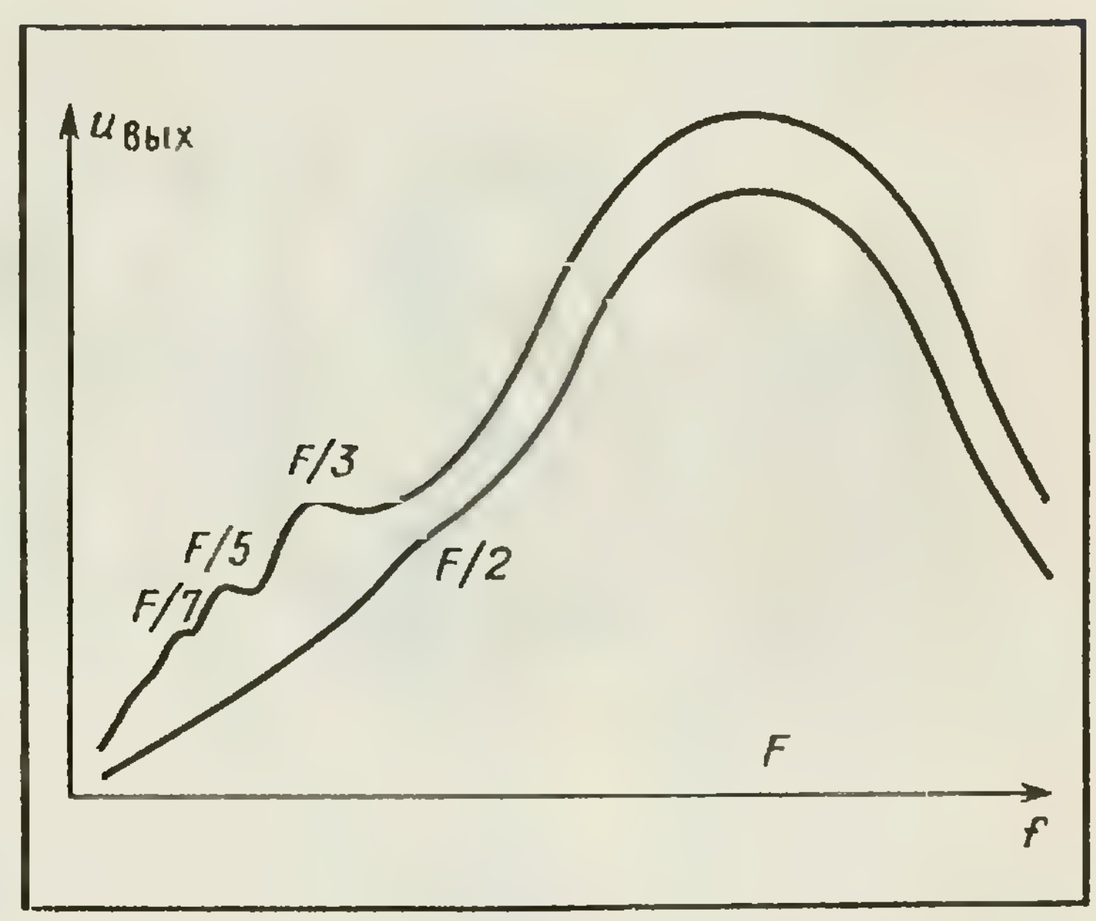


Рис.7

комплектации своих БК полезными самоделками.

В качестве примера мы приводим здесь частотные характеристики активного резонансного фильтра, который обычно применяется в схемах многополосных эквалайзеров или анализаторов спектра. Настройка всех десяти или двадцати фильтров в таких системах необходима, поскольку из-за разброса характеристик элементов попадание в заданные параметры (частота "резонанса", добротность, коэффициент передачи) маловероятно. БК легко научить не только "рисовать" частотные характеристики (причем в любом, хоть в двойном логарифмическом масштабе), но и находить частоту резонанса, коэффициент передачи и даже добротность всех фильтров.

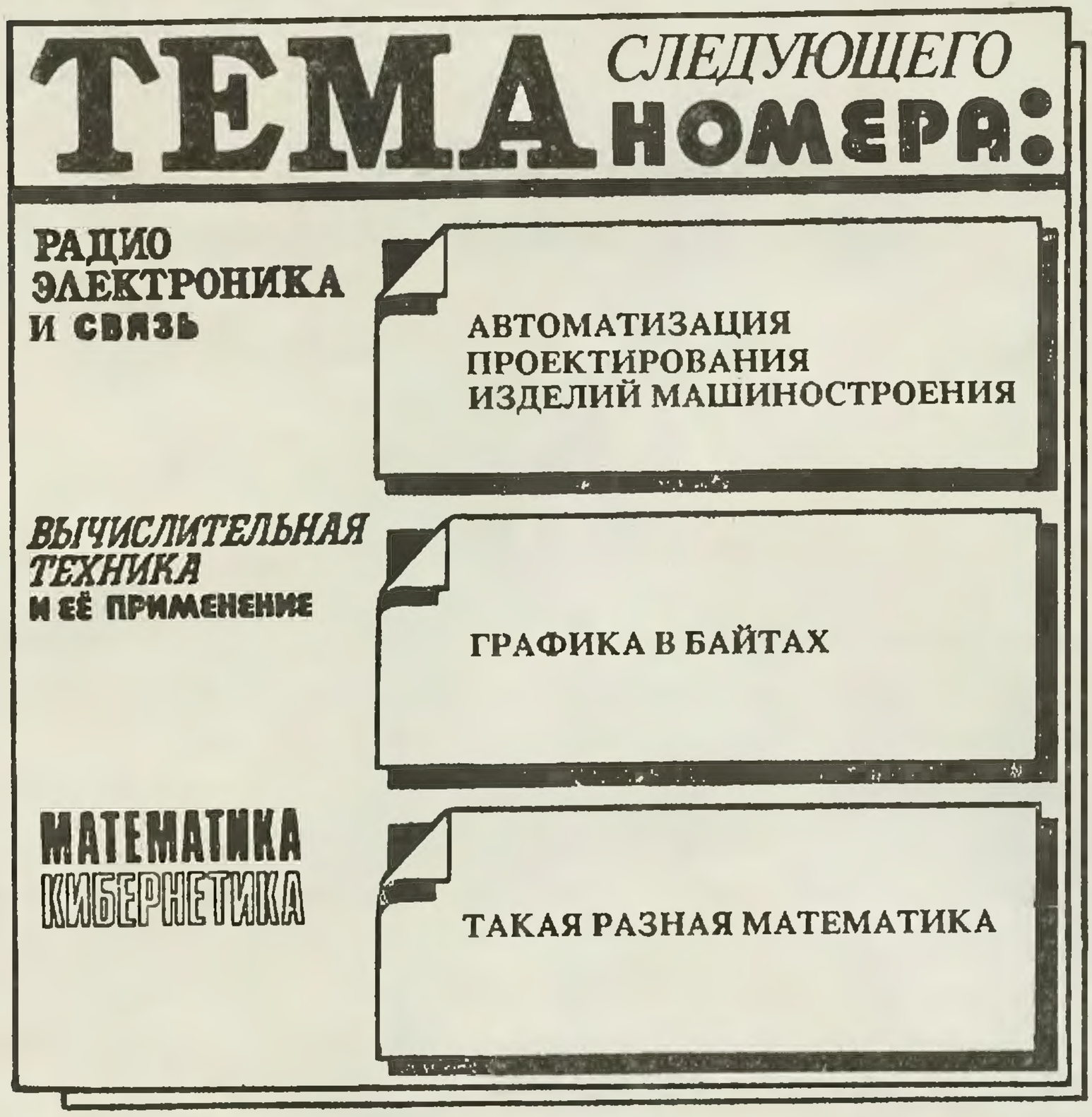
На приводимом нами рисунке (увы! — это снова лишь рисунок с экрана) изображен отклик фильтра на синусоидальное напряжение от нашего генератора и на меандр с той же амплитудой. Четко видны отклики на нечетные гармоники меандра (вплоть до седьмой) при отсутствии четных гармоник, что говорит о правильной скважности меандра. Легкая выпуклость на синусоидальном отклике в области, соответствующей второй гармонике (рис.7), означает небольшую асимметрию нашей псевдосинусоиды. Возможна настройка синус-преобразователя именно по минимуму этого пика. Наша же схема была настроена на глазок по снятой БК "осциллограмме". Видно, что такая настройка достаточна для целей наших измерений (отметим, что нечетные гармоники в синусоиде также практически отсутствуют).

И 27 Игра — дело серьезное. — М.: Знание, 1991. — 48 с. — (Новое в жизни, науке, технике. Сер. "Вычислительная техника и ее применение"; № 9).
ISBN 5-07-002178-8
35 к.

В редакции собран большой объем предложений по компьютерным играм. Имеется и очень большой (судя по письмам) спрос на подобную информацию. В этом выпуске выбраны наиболее интересные из грамотно и кондиционно представленных листингов игровых программ. Проверены только те, которые имеют заключение рецензента.

Материал рассчитан на широкий круг читателей.

2404040000 ББК 32.97



Научно-популярное издание

ИГРА — ДЕЛО СЕРЬЕЗНОЕ

Зам. главного редактора *Г. Г. Карвовский*
Редактор *Б. М. Васильев*
Мл.редактор *Н. А. Васильева*
Художник *В. Н. Конюхов*
Худож.редактор *И. А. Емельянова*
Техн.редактор *Т. В. Луговская*
Корректор *В. И. Гуляева*

ИБ № 11845

Подписано к печати 13.08.91. Формат бумаги 70x100[1]/16. Бумага офсетная. Печать офсетная. Усл.печ.л.3,90. Усл.кр.-отт.8,45. Уч.-изд.л.3,86. Тираж 49638 экз. Заказ 2374. Цена 35 коп. Издательство "Знание". 101835, ГСП, Москва, Центр, проезд Серова,д.4. Индекс заказа 914708. Отпечатано с оригинал-макета, подготовленного издательством "Знание" в издательской системе Xerox Ventura Publisher (ОС MS DOS), на ордена Трудового Красного Знамени Тверском полиграфическом комбинате Государственного комитета СССР по печати. 170024, г.Тверь, пр.Ленина,5.

INWESTSERVIS

Как быть, если одной персональной ЭВМ недостаточно, а цены на локальные сети кажутся непомерно высокими?

МЫ РЕШИМ ЭТУ ЗАДАЧУ ДЛЯ ВАС!

Львовское МП "Инвестсервис" предлагает

МНОГОТЕРМИНАЛЬНЫЙ КОМПЛЕКС КРАБ НА БАЗЕ ПЭВМ IBM PC AT/XT.

Комплекс КРАБ - это:

- полная совместимость с системами MS/DOS, PICK, PC/MOS, XENIX, MSM...
- комплектация лучшими отечественными терминалами, возможно подключение собственных аппаратных средств заказчика;
- многоканальные адаптеры, обеспечивающие подключение 4 — 16 терминалов.

Неплохо, правда? Но главный сюрприз впереди, когда вы обратитесь за дополнительной информацией и узнаете смехотворно низкую цену и заманчивые условия поставки.

КРАБ успешно применяют медики и нефтяники, филологи и шахтеры.

КРАБ покупают все - от крохотного кооператива до гигантского объединения.

Разработчики программного обеспечения, купившие КРАБ, получают право поставлять его пользователям вместе с собственными программными изделиями и вступить в ассоциацию "KRABUS".

ЗАПОМНИТЕ - ПРОФЕССИОНАЛЫ ВЫБИРАЮТ КРАБ!

Наш почтовый адрес:
290044, Львов-44, а/я 8863
МП "Инвестсервис"

Контактные телефоны:
35-35-79, 34-32-12 с 8 до 17 часов;
34-29-42 круглосуточно.

ИНВЕСТСЕРВИС

MONITOR

ВЫСОКОЕ КАЧЕСТВО!

Львовский НПК "Монитор" предлагает профессиональную автоматизированную систему проектирования РЭА.

Состав аппаратных средств:

- эмулятор однокристальных микро-ЭВМ серии К1816ВЕ48;
- эмулятор однокристальных микро-ЭВМ серии К1816ВЕ51
  - работают под управлением ПЭВМ типа IBM PC AT/XT.
- универсальные программаторы:
  модель А для программирования ППЗУ и ОЭВМ: К573РФ2, РФ5, РФ3, РФ4, РФ6, РФ8, К556РТ4-РТ7, РТ11-РТ18, К155РЕ3, i2764, КР1816ВЕ48
  - работает под управлением ПЭВМ типа IBM PC AT/XT и автономно под управлением собственной микро-ОЭВМ.
  модель В для программирования ППЗУ и ОЭВМ: К573РФ2, РФ5, РФ4, РФ6, РФ8, i2716, 2732, 2764, 27128, 27256, 27512, К155РЕ3, К556РТ4, К556РТ5, РТ11-РТ18, К1816ВЕ48, К1816ВЕ51, К1813ВЕ1, i8748, i8751, i2920
  - работает только под управлением ПЭВМ типа IBM PC AT/XT.

Состав интегрированных программных средств, работающих в среде MS DOS:

- пакеты программ поддержки функционирования аппаратных эмуляторов микро-ОЭВМ;
- пакеты программ поддержки функционирования универсальных программаторов ППЗУ и ОЭВМ;
- пакеты программных кросс-средств для ОЭВМ К1816ВЕ48, К1816ВЕ51.
  - классический набор, дружественный интерфейс, комфортное меню.

Конфигурацию САПР выбирает заказчик.
Имеются системные и игровые программы для ПК-01 "Львов".

НАШ ОПЫТ И РЕПУТАЦИЯ — ГАРАНТИЯ ВАШИХ УСПЕХОВ!

Обращаться по адресу: 290044, Львов, а/я 8863, НПК "Монитор".
Телефоны: 35-35-79 с 9 до 17 часов;
34-29-42 с 19 до 8 часов.

Цена 35 коп. Индекс 70195

Адрес подписчика:

Сол. 5-27

Издательство «Знание»

Подписная научно-популярная серия

ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА И ЕЁ ПРИМЕНЕНИЕ

Машины не смогут достичь человеческого уровня при решении наиболее интересных задач до тех пор, пока программы не станут похожими на людей в двух очень важных отношениях: способности накапливать свой собственный опыт на протяжении длительного времени, так сказать, "жизненный опыт", и способности общаться друг с другом, обучаясь таким образом

Дуглас Б. Ленат

Наш адрес:
101835,
Москва,
Центр,
проезд
Серова, 4